

# Parallel Computing - Notes - v0.2.0-dev

260236

October 2024

## Preface

Every theory section in these notes has been taken from the sources:

- Course slides. [\[1\]](#)

About:

 [GitHub repository](#)

These notes are an unofficial resource and shouldn't replace the course material or any other book on parallel computing. It is not made for commercial purposes. I've made the following notes to help me improve my knowledge and maybe it can be helpful for everyone.

As I have highlighted, a student should choose the teacher's material or a book on the topic. These notes can only be a helpful material.

## Contents

<b>1</b>	<b>PRAM</b>	<b>4</b>
1.1	Prerequisites . . . . .	4
1.2	Definition . . . . .	4
	<b>Index</b>	<b>6</b>

# 1 PRAM

## 1.1 Prerequisites

Before we introduce the PRAM model, we need to cover some useful topics.

- A **Machine Model** describes a “machine”. It gives a value to the operations on the machine. It is necessary because: it makes it easy to deal with algorithms; it achieves complexity bounds; it analyses maximum parallelism.
- A **Random Access Machine (RAM)** is a model of computation that describes an abstract machine in the general class of register machines. Some features are:
  - **Unbounded** number of local memory cells;
  - Each memory cell can hold an integer of **unbounded** size;
  - Instruction set includes simple operations, data operations, comparator, branches;
  - All operations take **unit time**;
  - The definition of **time complexity** is the number of instructions executed;
  - The definition of **space complexity** is the number of memory cells used.

---

## 1.2 Definition

### Definition 1: PRAM

A **parallel random-access machine (parallel RAM or PRAM)** is a **shared-memory abstract machine**. As its name indicates, the PRAM is intended as the parallel-computing analogy to the random-access machine (RAM) (not to be confused with random-access memory). In the same way that the RAM is used by sequential-algorithm designers to model algorithmic performance (such as time complexity), the **PRAM is used by parallel-algorithm designers to model parallel algorithmic performance** (such as time complexity, where the number of processors assumed is typically also stated).

The PRAM model has many interesting features:

- **Unbounded collection of RAM processors** ( $P_0$ ,  $P_1$ , and so on);
- Processors don't have tape;
- Each processor has **unbounded registers**;
- **Unbounded collection of share memory cells**;
- All **processors can access all memory cells in unit time**;
- All **communication via shared memory**.

## References

- [1] Ferrandi Fabrizio. Parallel computing. Slides from the HPC-E master's degree course on Politecnico di Milano, 2024.

## Index

### **M**

Machine Model 4

### **P**

Parallel Random-Access Machine (parallel RAM or PRAM) 4

### **R**

Random Access Machine (RAM) 4