

# Parallelism Module

## Concurrency Parallelism & Distributed Systems

Carlos Segarra

December 19, 2019

## Contents

<b>1</b>	<b>Understanding Parallelism</b>	<b>2</b>
1.1	Definition of a Distributed System . . . . .	2
1.2	Challenges of Distributed Systems . . . . .	2
<b>2</b>	<b>OpenMP</b>	<b>2</b>
<b>3</b>	<b>MPI: Message Passing Interface</b>	<b>2</b>
<b>4</b>	<b>Programming with CUDA</b>	<b>2</b>
4.1	GPU Architecture . . . . .	2
4.2	CUDA: Devices, Kernel Definitions, and Offloading . . . . .	3
4.3	CUDA: Blocks, Threads, and Indexing . . . . .	3
4.4	CUDA: Accessing (global) Memory . . . . .	3
4.5	CUDA: Cooperating Threads and Shared Memory . . . . .	3

# 1 Understanding Parallelism

## 1.1 Definition of a Distributed System

## 1.2 Challenges of Distributed Systems

# 2 OpenMP

# 3 MPI: Message Passing Interface

# 4 Programming with CUDA

## 4.1 GPU Architecture

### CPU vs GPU Architecture:

- CPUs are designed for general purpose computing.
- GPUs are specialized for highly-parallel, compute-intensive computation.
- Large Caches to reduce impact of long latency.
- Massive number of threads.

The testbed we use in CPDS is the MinoTauro cluster:

Listing 1: Output of running `nvidia-smi` in the cluster.

Thu Dec 19 11:34:34 2019

NVIDIA-SMI		418.67		Driver Version: 418.67		CUDA Version: 10.1	
GPU	Name	Persistence-M		Bus-Id	Disp.A	Volatile	Uncorr. ECC
Fan	Temp	Perf	Pwr:Usage/Cap	Memory-Usage		GPU-Util	Compute M.
0	Tesla K80		Off	00000000:04:00.0	Off		0
N/A	43C	P0	59W / 149W	0MiB / 11441MiB		0%	Default
1	Tesla K80		Off	00000000:05:00.0	Off		0
N/A	37C	P0	73W / 149W	0MiB / 11441MiB		0%	Default
2	Tesla K80		Off	00000000:85:00.0	Off		0
N/A	35C	P0	64W / 149W	0MiB / 11441MiB		0%	Default
3	Tesla K80		Off	00000000:86:00.0	Off		0
N/A	44C	P0	76W / 149W	0MiB / 11441MiB		92%	Default
Processes:						GPU Memory	
GPU	PID	Type	Process	name		Usage	
No running processes found							

- 4.2 CUDA: Devices, Kernel Definitions, and Offloading
- 4.3 CUDA: Blocks, Threads, and Indexing
- 4.4 CUDA: Accessing (global) Memory
- 4.5 CUDA: Cooperating Threads and Shared Memory