CS 220: Introduction to Parallel Computing

Non-Blocking Communication

Lecture 17

Today's Agenda

- Review: Measuring Performance
- Non-Blocking Communication
- Project 2

Today's Agenda

- **Review: Measuring Performance**
- Non-Blocking Communication
- Project 2

Measuring Parallel Performance

- As we discussed in the previous class, there are two common metrics for measuring parallel performance:
  - Speedup
  - Parallel Efficiency
- We also have to consider **scalability**

Speedup

- The **speedup** of a parallel program is given by:

$$S = \frac{T_{serial}}{T_{parallel}}$$

- How long the serial (original, non-parallel) program takes divided by the parallel run time
- Best speedup possible: ***S = p***
  - Where p is the number of processes

Efficiency

- The **parallel efficiency** of a program is given by:

$$E = \frac{S}{p} = \frac{T_{serial}}{pT_{parallel}}$$

- The speedup divided by the number of processes
- Best efficiency possible: **1**

Does it Scale?

Processes	1	2	4	8	16
Time (s)	300	150	75	50	40

Let's calculate:

- Speedup
- Efficiency

Today's Agenda

- Review: Measuring Performance
- **Non-Blocking Communication**
- Project 2

Blocking or Non-Blocking?

- We've made extensive use of MPI_Send and MPI_Recv
- These are both **blocking** operations
- The program won't move on to the next instruction until the function call completes
- If we're sending and nobody hears our cry, then we'll just sit there and wait
  - Same thing for receiving…

Non-Blocking Operations

- Non-blocking functions are prefixed with an "I"
- MPI_Iprobe is our first: it checks to determine if a message is waiting for us or not
  - If there is no message available, it just returns immediately
- MPI_Isend
- MPI_Irecv

Getting Everyone to Stop

- In an MPI program, all the processes work independently unless they're passing messages
- If we use blocking operations, the processes will stop
- In P2, we'll be using non-blocking send/receive operations

Today's Agenda

- Review: Measuring Performance
- Non-Blocking Communication
- **Project 2**

Project 2

- P2 is now available on the course website!
- We'll be using MPI to speed up a particular type of search algorithm…
- Let's walk through this now