Documentation Laboratory 9

Hiticas Claudiu Emanuel
Group 924

The goal of this lab is to implement a distributed algorithm using MPI.

Requirement

Perform the multiplication of 2 polynomials, by distributing computation across several nodes using MPI. Use both the regular O(n2) algorithm and the Karatsuba algorithm. Compare the performance with the "regular" CPU implementation from lab 5. *Bonus: do the same for the multiplication of big numbers*

Documentation

Implementation in C++ using OpenMPI 3.0. The performance is measured on my personal computer with the following config:
MacBook Air (13-inch, Early 2015)
Processor: 1.6 GHz Dual-Core Intel Core i5
Memory: 8 GB 1600 MHz DDR3
Graphics: Intel HD Graphics 6000 1536 MB

Algorithm

Naive approach

The algorithm I decided to use is the following: Each worked gets a chunk of the end result to comute. Let a and b - the polynoms to multiply. Now, let m = a * b. Each worked node will have to compute m[st:fn] - the result on the positions between st and fn It's easy to see that because each worked will need to compute m[st:fn] it only needs the values a[0:fn] and b[0:fn].

Karatsuba

P1 * P2 = (P11 + P12 + ... + P1k) * P2 I just split the first polynom and distribute the work equally. The master does it's part, then sums up all the result.
master: assign chunks to all of the slave nodes
master: sends the chunks to the slaves
master: keeps a chunk for him & solve
master: wait for the results
master: adds up the results together

slave: get the chunk assigned from master
slave: solve the subproblem
slave: send the result back to the master

Distribution and Communication

All the communication is done throgh the OpenMPI 3.0 library using specific methods such as MPI_BSend(), MPI_SSend(), MPI_Recv(). There is only one master which splits the work among the slave (worker nodes). Since we don't want the master to wait for the results, the master will do slave work after assigning everything to the other nodes.

Performance measurements

The best one, because mpi did that for us!