Parallel merge-sort algorithm using MPI

Group Members:

A.S.K Viswas(AM.EN.U4CSE20012)

Atla Suraj Reddy(AM.EN.U4CSE20013)

Nandipati Sai Sandeep(AM.EN.U4CSE20049)

Padala Laxmi Praneeth(AM.EN.U4CSE20052)

CGS Pranav Advaith(AM.EN.U4CSE20056)

ABSTRACT

Parallel computing is a sort of computation in which various tasks or processes are run at the same time. In contrast, distributed computing is that type of computing in which the components are located on various networked systems that interact and coordinate their actions by passing messages to one another. Merge sort is a divide and conquer paradigm. It divides those issue under several sub problems that are littler instances of the same problem.

Parallel merge-sort is a sorting algorithm that uses the divide-and-conquer strategy to sort a large array of elements in parallel. It divides the array into subarrays and distributes them among different processes using MPI (Message Passing Interface). Each process sorts its subarray using a sequential merge-sort algorithm and then sends it back to the parent process. The parent process merges the sorted subarrays using a merge function until the whole array is sorted.

Since our main aim is sort them, the thing is just single instruction on multiple data. So, it is a good idea to use single instruction multiple parallel implementation.

We also compare measured time of serial and parallel version of merge sort, parallel version has achieved best speedup. Parallel merge-sort can achieve a speedup over serial merge-sort by reducing the communication and computation costs.