Assignment 2

Basic MPI implementations

There are various types of send and receive functions implemented in MPI. Some of them are tried and discussed below.

Blocking Asynchronous Send

In blocking send, function returns when it is safe to change the application buffer for reuse. Now, in blocking asynchronous send, it states that data will eventually be delivered to the receiver. We will not observe deadlock in this case. Syntax for blocking asynchronous send is:

MPI_Send (&buf, count, datatype, dest, tag, comm)

Blocking Synchronous Send

In blocking send, function returns when it is safe to change the application buffer for reuse. Now, in blocking Synchronous send, handshake takes place with the receive task to confirm a safe send. Hence we can observe **deadlock** in this case when 2 processors are first sending data to each other and then receiving data from each other. Syntax for blocking synchronous send is:

MPI_Ssend (&buf, count, datatype, dest, tag, comm)

Unblocking Send and Receive

These unblocking send and receive returns immediately. They do not wait for any communication events to complete. Wait function calls are required after these function calls to insure that all unblocking send and receive are completed.

MPI_Isend

Identifies an area in memory to serve as send buffer and starts immediately without waiting for the message to be copied out in the send buffer. A care need to be taken that the application buffer shouldn't be changed until it is confirmed that send has occurred. Syntax for the same is:

MPI_Isend (&buf, count, datatype, dest, tag, comm, &request)

MPI_Irecv

Identifies an area in memory to serve as receive buffer and processing continues immediately without actually waiting for the message to be received and copied into the application buffer. Syntax for the same is:

MPI_Irecv (&buf, count, datatype, source, tag, comm, &request)