

Comunicación bloqueante y no bloqueante

Comunicación bloqueante:

Una comunicación bloqueante se da cuando se hace uso de las funciones MPI_Send() y MPI_RECV. Estas funciones no devuelven hasta que la comunicación se haya finalizado. Esto quiere decir que el búfer pasado a MPI_Send puede ser reutilizado. Del mismo modo, MPI_Recv retorna cuando el búfer de recepción se ha llenado con datos válidos. La comunicación bloqueante se utiliza más a menudo, ya que es más fácil de usar. [1]

Ejemplo:

```
if(rank==0)
{
    MPI_Send(x a proceso 1)
    MPI_Recv(y desde proceso 1)
}
if(rank==1)
{
    MPI_Send(y a proceso 0);
    MPI_Recv(x desde proceso 0);
}
```

En este ejemplo se hace uso de una comunicación bloqueante:

- 1) Proceso 0 envía x al proceso 1 y se bloquea hasta que el proceso 1 reciba x.
- 2) Proceso 1 envía y al proceso 0 y se bloquea hasta que el proceso 0 reciba y.
- 3) Pero el proceso 0 y 1 están bloqueados, provocando que se queden bloqueados infinitamente hasta que ambos procesos mueran.

Comunicación no bloqueante:

La comunicación no bloqueante se realiza mediante las funciones MPI_Isend() y MPI_Irecv. Estas funciones retornan de inmediato incluso si la comunicación no ha terminado todavía. Se debe hacer uso de las funciones MPI_Wait() y MPI_Probe() para verificar si la comunicación ha finalizado. Esta comunicación se utiliza cuando es realmente necesaria. [1]

Ejemplo:

```
if(rank==0)
{
    MPI_Isend(x a proceso 1)
    MPI_Irecv(y desde proceso 1)
}
if(rank==1)
{
    MPI_Isend(y a proceso 0);
    MPI_Irecv(x desde proceso 0);
}
```

Christian Flores Meléndez

En este ejemplo se hace uso de una comunicación no bloqueante:

A diferencia de lo que sucede en el ejemplo de la comunicación bloqueante, en este ejemplo ni uno de los dos procesos se quedan bloqueados infinitamente, ya que las funciones `MPI_Isend()` y `MPI_Irecv()` no esperan y retornan inmediatamente.

Referencias:

- [1] <<unlp>> 01 11 2008. [En línea]. Available:
<http://electro.fisica.unlp.edu.ar/arq/laboratorios/cluster/tutorial.MPI.basico.pdf> [Último acceso: 2016 08 31]