

Concorrenca e Paralelismo. Bloque II Paralelismo

Práctica 2: colectivas MPI en la cuenta de las apariciones de una letra

Primavera 2023



Colectivas MPI en la cuenta de las apariciones de una letra

Paralelización realizada en la práctica 1 + mejoras práctica 2!

- Implementación SPMD
- La E/S (scanf/printf) la hace el proceso 0
- Distribuir n y L a todos los procesos (con Send/Recv) **Ahora con operación colectiva MPI!**
- Reparto de la carga de trabajo en el bucle for con “paso”
 $i += \text{numprocs}$ en lugar de $i++$
- Recoger número de apariciones detectado por cada proceso (con Send/Recv) **Ahora con operación colectiva MPI!**

Colectivas MPI en la cuenta de las apariciones de una letra

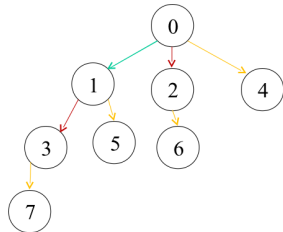
Uso de colectivas MPI

- Inicialmente operaciones colectivas estándar de MPI
- Implementación de colectiva en árbol binomial, implementación que denominaremos MPI_BinomialColectiva, a utilizar **SOLO** en la distribución de n y L .
- Posteriormente introducción de implementación propia de colectiva **SOLO** para la recolección de *count*, inicialmente utilizando las mismas operaciones de Send/Recv que en la implementación sin colectivas (bucle for de Recv), implementación que denominaremos MPI_FlatticeColectiva. Asumir que la operación a realizar será una suma. El resto de parámetros de la cabecera deben ser los mismos que los de la colectiva estándar de MPI (incluido controlar el error).

Colectivas MPI en la cuenta de las apariciones de una letra

Implementación de Bcast con árbol binomial (MPI_BinomialBcast):

- Mismos parámetros que MPI_Bcast (consultar página man de MPI_Bcast para obtener cabecera), asumiendo por simplicidad que el root es el 0
- En el paso “i” los procesos con $myrank < 2^{i-1}$ se comunican con el proceso $myrank + 2^{i-1}$



Paso 1: 0 → 1

Paso 2: 0 → 2, 1 → 3

Paso 3: 0 → 4, 1 → 5, 2 → 6, 3 → 7