

Computación Paralela y Distribuida

2022-I

José Fiestas

26/05/22

Universidad de Ingeniería y Tecnología
jfiestas@utec.edu.pe

Práctica Dirigida 05:

MPI+OMP

Puntaje: 5 pts.

Ejercicio 1: Adivina mi número con MPI (1.5 pt)

Genere un modelo en MPI que simule la siguiente situación: Paolo está con 5 amigos. Él piensa un número entre 1 y 100, y dice a sus amigos: '¡Adivinen mi número!'. Sus amigos deben escribir el número en un papel y mostrarlo simultáneamente. Si logran todos adivinar el número en el mismo intento, Gabriel los invita a cenar al Restaurante Central (el más caro de Lima). Paolo les da 1000 intentos para hacerlo. Realice los experimentos con el modelo generado y compruebe si se acercan a la probabilidad de éxito según la teoría estadística.



Ejercicio 2: Arrays con OMP (1.5 pt)

Dada la siguiente función que realiza operaciones con los arrays A, B y C:

```
double OpArr(double A[], double B[], double C[], int n){
    int i, j;
    double s1, s2, a, res;
    calculo_a(A,n); /* funcion calculo a */
    calculo_b(B,n); /* funcion calculo b */
    calculo_c(C,n); /* funcion calculo c */
    for (i=0; i<n; i++) { /* primer bucle for*/
        s1=0;
        for (j=0; j<n; j++) s1+=A[i]*B[i];
        for (j=0; j<n; j++) A[i]*=s1;
    }
    for (i=0; i<n; i++) { /* segundo bucle for */
        s2=0;
        for (j=0; j<n; j++) s2+=B[i]*C[i];
        for (j=0; j<n; j++) C[i]*=s2;
    }
    /* calculo final */
    a=s1/s2;
    res=0;
    for (i=0; i<n; i++) res+=a*C[i];
    return res;
}
```

Ejercicio 2: Arrays con OMP (cont.)

Considere que las funciones *calculo_a*, *calculo_b*, y *calculo_c*, modifican A, B, C respectivamente

- a) Construya el DAG de la función OpArr()
- b) Identifique las tareas y sus dependencias de acuerdo al DAG y paralelice la función con OMP. Utilice task con dependencias. Asuma que los bucles no necesitan ser paralelizados
- c) Determine la complejidad del algoritmo secuencial, y el paralelo, así como el speedup. Asuma que las funciones *calculo_a*, *calculo_b*, y *calculo_c* tienen complejidad $O(2n \log^2(n/p))$ flops cada una.

Ejercicio 3: Multiplicación matriz-vector con MPI+OMP (2 pts)

Considere el código de multiplicación matriz-vector de la PD4 (Ej.3).

- a) Genere una gráfica tiempo de ejecución vs. número de procesos en MPI, para $np=2,4,8,16$
- b) Paralelice la multiplicación matriz-vector con OMP, y genere una gráfica tiempo de ejecución vs. número de threads, para $nthreads=2,4,8,16$
- c) Paralelice la multiplicación matriz-vector con MPI+OMP, y compare en una gráfica velocidad vs. número de procesos, las curvas para $np=4,8,16$ y $nthreads = 4,8$, con los correspondientes modelos en MPI ($np=4,8,16$)
- d) Discuta la escalabilidad de los resultados y decida sobre la utilidad de implementar un modelo híbrido para este problema

Compile y ejecute este análisis en el cluster Khipu. Incluya los scripts usados y describa los parámetros asignados para procesos MPI e hilos OMP