

Práctica 4

Paralelismo en sistemas de memoria distribuida

Objetivos:

- Aprender a paralelizar una aplicación en un sistema de memoria distribuida utilizando técnicas de paso de mensajes.
- Estudiar el [API](#) de [MPI](#) y aplicar distintas estrategias de paralelismo en su aplicación.
- Aplicar métodos y técnicas propios de esta asignatura para estimar las ganancias máximas y la eficiencia del proceso de paralelización.

Desarrollo:

En esta práctica se va a continuar con el problema tratado en la práctica anterior. En este caso se desarrollará la práctica en múltiples computadores (**mínimo 3**), de tal forma que se haga la comunicación utilizando técnicas de paso de mensaje con MPI y estrategia que más se adecue al problema particular de cada grupo.

Pasos para compilar y ejecutar (actualizado a noviembre de 2018):

1. Todas las máquinas deben tener el **mismo usuario**. Todos los ejecutables deben tener el **mismo nombre** en cada máquina y estar en la **misma carpeta**.
2. Compilar: `mpicc programa.c -o programa`. Recuerde que el programa, junto con sus dependencias, deben copiarse y ser accesibles en cada nodo de su multicomputador.
3. Crear archivo de hosts con las IPs de cada máquina de la red y copiar al resto de máquinas.
Archivo de hosts: archivo de texto que especifica el nombre o la IP de las máquinas que forman el supercomputador. Cada nombre/IP en una línea.
4. Cree un certificado de clave pública ssh en uno de los nodos:
`ssh-keygen -t rsa (crea la carpeta oculta ~/.ssh)`
5. Copiar el archivo `~/.ssh/id_rsa.pub` en la carpeta `~/.ssh` del resto de máquinas y cambiar su nombre a `authorized_keys`
6. Ejecutar el programa en la máquina que se creó el certificado:

```
mpirun - hostfile <archivo_hosts> -n <núm_procesos>
./programa
```

```
#include <stdio.h>
#include <stdlib.h>
#include <mpi.h>

// Ejemplo 'Hola' donde cada procesador
// se identifica
int main(int argc, char **argv)
{
    int id, nprocs;

    MPI_Init(&argc,&argv);
    MPI_Comm_size( MPI_COMM_WORLD, &nprocs ); // Devuelve el numero de procesos
                                              // en el COMM_WORLD (comunicador)
    MPI_Comm_rank( MPI_COMM_WORLD, &id );    // Identifica el id asignado

    printf("Hola. Soy el procesador %d de un total de %d\n", id, nprocs);

    if (id == 1) // Selección de procesamiento en el procesador 1
    {
        // Solo se ejecuta en el procesador 1
        printf("\nHola desde el procesador %d\n", id);
    }

    MPI_Finalize();

    return 0;
}
```

```
$ mpirun -hostfile hosts -n 4 ./Ejemplo1
Hola. Soy el procesador 2 de un total de 4
Hola. Soy el procesador 1 de un total de 4
Hola. Soy el procesador 0 de un total de 4
Hola. Soy el procesador 3 de un total de 4

Hola desde el procesador 1
```

Observaciones:

- Usar el mismo directorio en todas las máquinas: P.e. Carpeta personal (\$HOME)
- Abrir sesión con **el mismo usuario** en todos los computadores
- Puede ser necesario modificar los permisos de los archivos:
 - Archivo de hosts: `chmod 600 <archivo_hosts>`
 - Archivo ejecutable para permitir ejecución remota: `chmod o+x programa`

Notas generales a la práctica:

- La implementación realizada tendrá que poder ejecutarse bajo el sistema operativo Linux del laboratorio
- Se debe calcular la ganancia en velocidad y la eficiencia obtenida
- La nota de la práctica tendrá en cuenta tanto la funcionalidad como el rendimiento conseguido. Se realizarán pruebas de rendimiento en una máquina del laboratorio para valorar qué grupo ha conseguido mayor rendimiento. Los mejores grupos tendrán un incremento en su nota de: mejor tiempo: 2 puntos, segundo mejor tiempo: 1 punto, tercero: 0.5 puntos.
- Las/los estudiantes entregarán, además de la aplicación desarrollada, una memoria, estructurada según indicaciones del profesor, con la información obtenida.
- **Es obligatorio** entregar un *Makefile* con las reglas oportunas para compilar y limpiar su programa (`make clean`) de manera sencilla.
- **Entrega:** la semana del 17 de diciembre.
- Los trabajos teórico/prácticos realizados han de ser originales. La detección de copia o plagio supondrá la calificación de "0" en la prueba correspondiente. Se informará la dirección de Departamento y de la EPS sobre esta incidencia. La reiteración en la conducta en esta u otra asignatura conllevará la notificación al vicerrectorado correspondiente de las faltas cometidas para que estudien el caso y sancionen según la legislación (Reglamento de disciplina académica de los Centros oficiales de Enseñanza Superior y de Enseñanza Técnica dependientes del Ministerio de Educación Nacional BOE 12/10/1954).