

Escuela de Ciencias de la Computación

4ta Práctica de Laboratorio

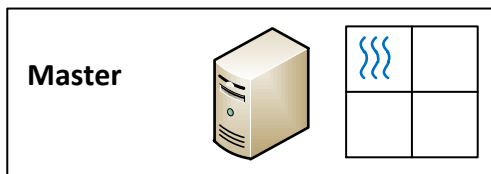
CC462 Sistemas Concurrentes y Distribuidos

Escalabilidad Dinámica.

Desarrollar un Programa con Escalabilidad Dinámica, se define como un programa que realiza una tarea muy grande en el cual se pueda distribuir la tarea en una red en tiempo de ejecución sin que la aplicación maestra se detenga.

La aplicación Maestra se encuentra en un Servidor en el que realiza la tarea por ejemplo el cálculo de una integral con una cantidad considerable de máximos locales y mínimos locales

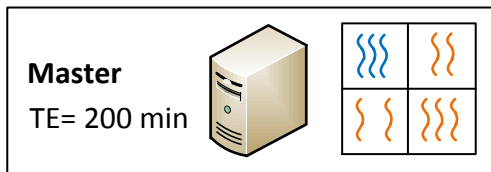
Asumiendo que cada máquina tenga 4 núcleos, la aplicación principal se ejecutara con un grupo m de hilos, esta incluye el ingreso de la función y los parámetros al programa para aplicar la función, si esta pc muere no se podrá evaluar el resultado posterior final. Los hilos azules representan el programa principal.



Los valores iniciales son la función, del rango a y b del rango de la integral y los intervalos, los valores de deben superar 2147483647 hasta un máximo de 9223372036854775807

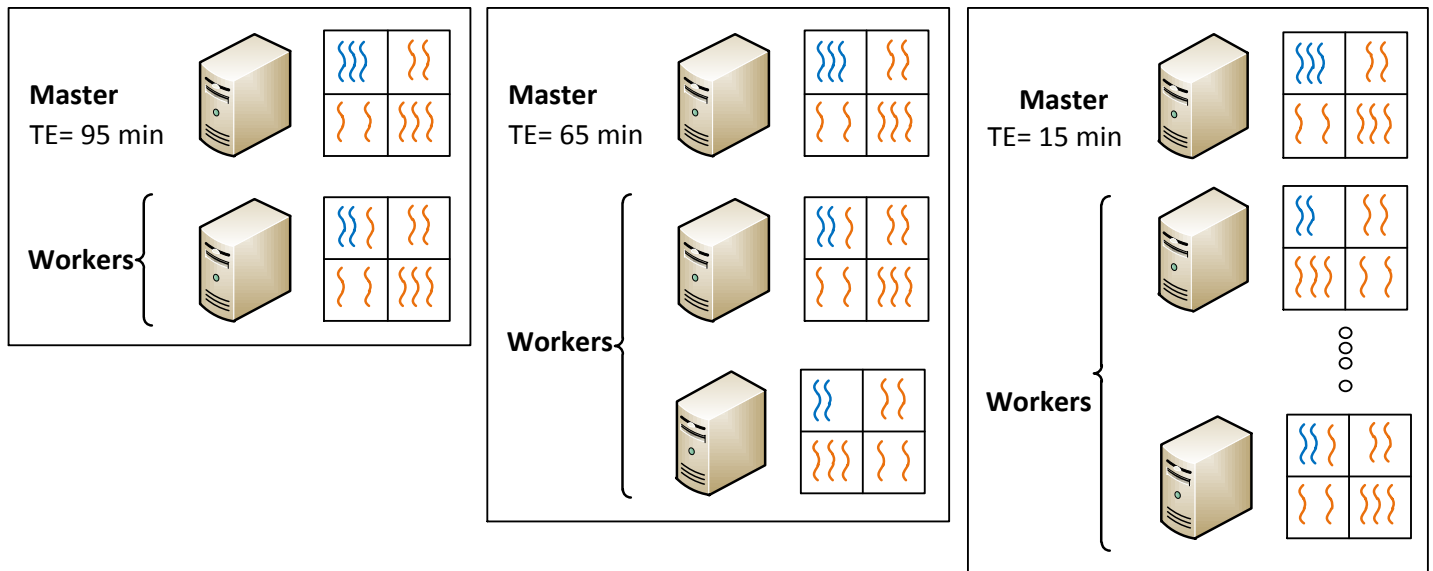
Caso 1

El Maestro puede ejecutar el programa sin los Workers, donde toda la aplicación se pueda ejecutar en el Maestro para esto asegurarse que todos los hilos del maestro se ejecuten en todos los núcleos, utilizando así todos los recursos. El programa debe de calcular el tiempo de ejecución en tiempo real con un indicador. Los hilos de color naranja son el trabajo específico en resolver la integral.



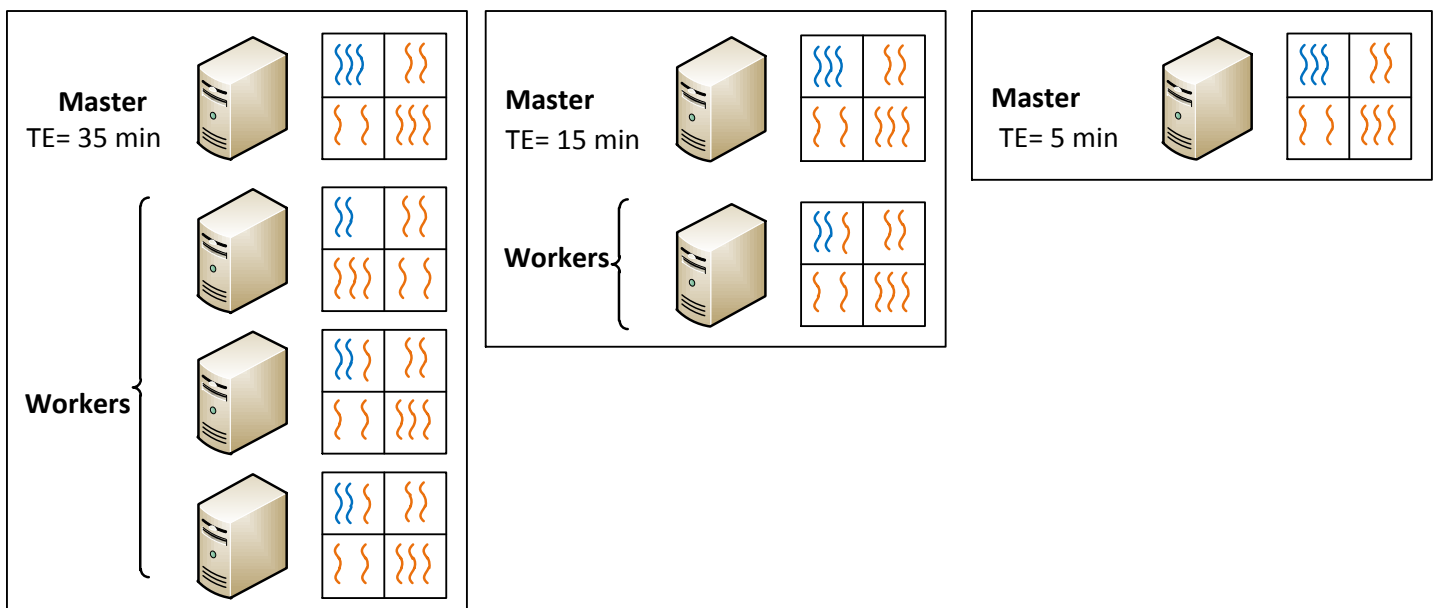
Caso 2

El Maestro puede iniciar independiente y este puede administrar a nuevos trabajadores que estén desde el inicio o nuevos trabajadores que ingresen en tiempo de ejecución, conforme se añadan. La añadidura de más nodos Workers implica que el tiempo de ejecución cambie conforme se redistribuyan el trabajo de los intervalos de la Integral. Al igual que el Maestro cada Worker tiene hilos azules que están encargados de la comunicación y administración del trabajo de la integral. Los hilos naranjas son las que ejecutan las tareas del trabajo específico de resolver la integral en cada Worker.



Caso 3

El maestro se percata la pérdida de un Nodo Worker. Cuando se pierde un Worker el maestro evaluar el intervalo de la integral perdida y lo redistribuye entre los demás nodos Workers que siguen activos. Donde como mínimo puede quedar el nodo Maestro. Cuando un Worker se pierda se tiene que revalorar el tiempo de ejecución de la tarea general.

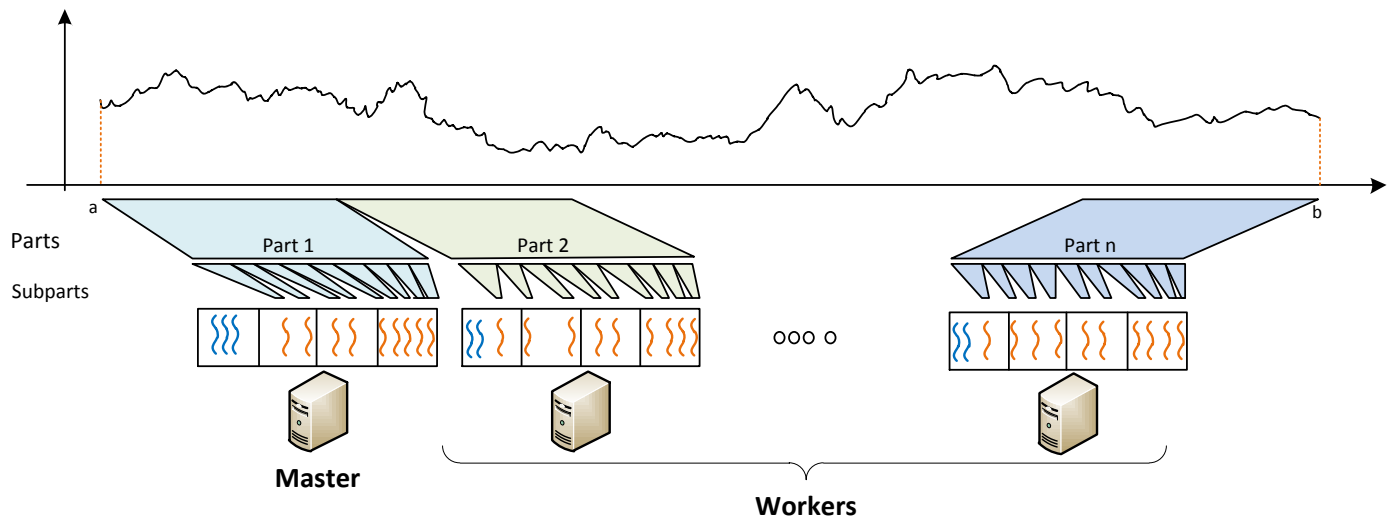


Caso 4

Sería la combinación y las permutaciones de los Casos 1, Casos 2, Casos 3.

Redistribución de Trabajo.

Buscar el número óptimo de hilos óptimo que se puede generar en una pc de acuerdo al número de núcleos. Donde cada uno tiene que realizar un tramo de la función y así repartirlo en hilos dentro de cada núcleo. Sabiendo que en tiempo de ejecución puede aumentar o disminuir hilos, como también perder intervalos de proceso por la salida o perdida de alguna pc, y así actualizar el trabajo de cada hilo existente y actualizarse el tiempo de ejecución.



Cada ítem es requisito para el siguiente.

1. Realizar el Maestro ingresando los parámetros iniciales y Caso 1 (pts3)
2. Realizar el Maestro y Workers Caso 2 (pts 3)
3. Realizar el Maestro y Workers Caso 3 (pts 4)
4. Realizar el Maestro y Workers Caso 4 (pts 4)
5. Realizar y validar la Redistribución de Trabajo. (pts 6)