

Descomposición de dominio

Se usa en estructuras de datos de gran tamaño. Se hace en dos fases:

1. **Troceo de datos:** Se divide la estructura de datos en subdominios más pequeños, centrándose en la estructura más grande o la más utilizada. De esta forma, se facilita el manejo de los datos y se reduce la complejidad del problema.
2. **Asociación de la computación a cada trozo**

Por ejemplo, para multiplicar una matriz por un vector (suponemos que $\text{filas} \% \text{procesos} = 0$). Cada una de las tareas calculará columnas/procesos filas de la matriz.

La regla del propietario nos dice que la tarea a la que se le asigna un dato es responsable de realizar todos los cálculos asociados al mismo. No siempre es posible de aplicar en todas las situaciones. En algunos casos, puede haber datos que necesiten ser procesados por varias tareas, lo que puede complicar la asignación de responsabilidades y hacer que la aplicación de la regla del propietario sea más difícil. Puede ser:

- **Centrada en la entrada:** todas las computaciones que usen los datos de entrada serán realizadas por la tarea a la que se le asigna la entrada. Esto significa que la tarea será responsable de realizar todas las operaciones necesarias para procesar los datos de entrada y producir los resultados necesarios.
- **Centrada en la salida:** la salida es computada por el proceso al que se le asigna esa salida. Esto significa que la tarea será responsable de realizar todas las operaciones necesarias para producir los resultados y entregarlos al proceso que solicita la salida.

Descomposición funcional

Consiste en descomponer el cálculo en tareas según las partes diferenciadas. Se hace en dos pasos también:

1. **Identificación de las fases del cálculo**
2. **Asignación de tareas a cada fase**

El grafo resultante suele tener forma de pipeline, es decir, una serie de pasos o procesos interconectados que se llevan a cabo en secuencia para procesar datos de manera eficiente y automatizada.

Por ejemplo, para calcular una matriz por un vector por el vector ordenado. Se multiplica el vector por la matriz y se crea el vector ordenado de forma paralela. Una vez listo se hace el producto escalar. Cada una de las tareas se podría descomponer.

Descomposición recursiva

Se basa en hacer divide y vencerás.

1. Dividir el problema original en subproblemas más pequeños, de manera que cada subproblema sea lo suficientemente simple como para poder resolverse fácilmente.
2. Resolver recursivamente cada subproblema por separado, utilizando la misma técnica de descomposición.
3. Combinar los resultados de cada subproblema para obtener la solución del problema original.

Si un subproblema ha alcanzado un tamaño crítico, se puede obtener su solución parcial sin necesidad de descomponerlo aún más. Este enfoque se utiliza a menudo en problemas que tienen una estructura recursiva natural, como la ordenación de una lista de elementos.

Por ejemplo, para el cálculo de áreas bajo la curva de una función. Calculándola de forma geométrica tendríamos cierto error. Podemos descomponer la integral en dos partes y calcularlo de forma geométrica para reducir el error. Hacemos esto de forma recursiva hasta que el error sea lo suficientemente pequeño y luego sumamos todo.

Descomposición especulativa

La descomposición especulativa es una estrategia utilizada en ciertos problemas en los que no es posible dividirlos en tareas independientes, pero sí se pueden dividir funcionalmente en fases que dependen condicionalmente de los resultados de fases anteriores.

En esta estrategia, se inicia la ejecución de tareas condicionales excluyentes sin esperar a la finalización de las tareas de las que dependen para seleccionar una de ellas. Esto significa que se ejecutan diferentes tareas de forma especulativa, asumiendo que se cumplirá una determinada condición en función de los resultados parciales obtenidos.

Una vez que la tarea de selección ha finalizado, se puede elegir el resultado correcto de entre los obtenidos por las tareas ejecutadas de forma especulativa. En otras palabras, se toma una decisión basada en la condición evaluada y se descartan los resultados incorrectos o no deseados generados por las tareas especulativas.

Esta descomposición especulativa permite avanzar en la ejecución del problema, incluso cuando no se disponen de todos los resultados necesarios para tomar una decisión definitiva.

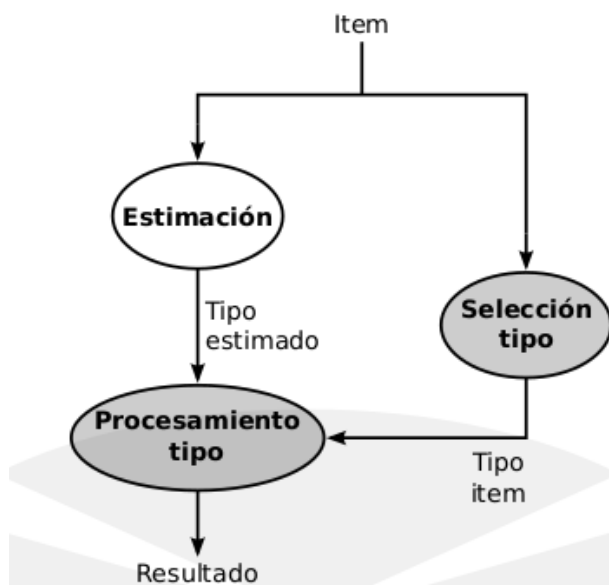
Al ejecutar las tareas de forma especulativa, se pueden obtener resultados parciales que ayudan a guiar la selección final una vez que se tenga la información completa.

Por ejemplo este problema se podría descomponer con descomposición especulativa:



$$T_{sec} \simeq T + T = 2T$$

El tipo de elemento tiene una tasa de acierto del 80%. Por tanto, si lo hacemos de forma especulativa:



- Aprovechamos la estimación eficiente del elemento.
- Se acierta en un 80% de los casos. En el resto, debe cancelarse la ejecución especulativa y comenzar el procesamiento correcto.
- El speed-up es cercano a 2, siempre que dispongamos de dos unidades de procesamiento.

$$T_{esp} \simeq 0.8T + 0.2(2T) = 1.2T$$