

Prólogo

Este trabajo se centra en el estudio y propuesta de técnicas de paralelización de bucles en programas secuenciales numéricos para sistemas multiprocesador y computadores vectoriales segmentados. Se hace especial hincapié en la existencia de recurrencias y sentencias condicionales en estos bucles.

Las técnicas de reestructuración descritas y presentadas se basan en un detallado análisis de las relaciones de precedencia determinadas por la ejecución secuencial del bucle. En cualquier caso, la reestructuración de código no debe afectar la semántica del programa original.

Se considera la evaluación de las características (paralelismo y longitud vectorial de las sentencias) que determinan el grado de concurrencia que se puede obtener del bucle original. Estas características se obtienen a partir del grafo de dependencias entre sentencias obtenido en tiempo de compilación y son una buena medida de la eficiencia del proceso de reestructuración realizado.

*Se propone un método, **Graph Traverse Scheduling**, que incorporado en un compilador permite obtener el máximo paralelismo del bucle. La distribución de operaciones e inserción de primitivas de sincronización realizada se basa en recorridos a través de un ciclo del grafo de dependencias que cumple unas determinadas características. Este método se ve soportado por otras técnicas que posibilitan la reestructuración de cualquier tipo de bucle.*