

Resumen

En programación funcional es común escribir los programas como la composición de funciones relativamente más sencillas. Esto tiene todas las ventajas asociadas a un estilo de programación modular. Sin embargo los programas obtenidos pueden ver afectada su eficiencia debido a la generación de las estructuras de datos intermedias que son utilizadas como comunicación entre las funciones involucradas en las composiciones.

Existe un conjunto de técnicas de transformación de programas, conocidas como *fusión* o *deforestación*, que apuntan a la eliminación de estas estructuras intermedias y permiten obtener definiciones más eficientes, equivalentes a las originales. El objetivo es posibilitar que el programador pueda continuar escribiendo sus programas de forma composicional y generar un código equivalente, más eficiente, que sea el que efectivamente se ejecute.

En este trabajo se considera el caso particular donde las funciones involucradas son tales que construyen sus resultados utilizando un parámetro adicional de acumulación. Las técnicas de fusión clásicas no suelen ser efectivas en el caso de estas funciones ya que no consiguen eliminar las estructuras intermedias cuando las mismas son generadas en los parámetros de acumulación.

Como primera aproximación a una solución del problema se realiza una revisión del operador *afold* propuesto por Pardo, el cual permite la definición de funciones con acumuladores por recursión estructural sobre tipos polinomiales. Se presenta una modificación al operador que permite ampliar el conjunto de funciones capaz de representar.

Posteriormente la tesis se concentra en la propuesta de un nuevo método de fusión basado en el método de “Short-Cut Fusion” que contempla el caso en que la función productora genera sus resultados utilizando parámetros de acumulación. Se muestra la aplicación del nuevo método a programas Haskell y se realiza una serie de pruebas que permiten comparar la performance desde el punto de vista del tiempo de ejecución y el espacio de memoria requerido entre las versiones originales y transformadas de un conjunto de programas ejemplo.