Investigación sobre la arquitectura de computadoras

Arquitecturas modernas emergentes de RISC y CISC

La conversión de instrucciones entre arquitecturas distintas, como CISC y RISC, representa un desafío importante en la compatibilidad de los sistemas modernos.

Las arquitecturas CISC cuentan con sus instrucciones complejas que pueden realizar varias operaciones en una sola línea de código.

Por otro lado, las arquitecturas RISC utilizan instrucciones más simples y uniformes que las CISC, y estas son diseñadas para mejorar la eficiencia y velocidad de ejecución.

Al traducir un programa de CISC a RISC, cada instrucción compleja debe descomponerse en varias instrucciones más simples, es decir, generando lo que se conoce como una inflación de instrucciones.

Esta inflación es una fuente principal de sobrecarga en los traductores binarios dinámicos, pues una sola instrucción original puede convertirse en múltiples instrucciones en la arquitectura destino.

**La arquitectura destino** se refiere al tipo de arquitectura de procesador en la que se va a ejecutar el código después de que se haya traducido.

Aunque existen técnicas para reducir esta inflación, en la cual todavía se presenta una pérdida considerable de rendimiento en muchas herramientas actuales.

Para analizar y mejorar este proceso, se ha desarrollado una herramienta llamada Deflater que evalúa la inflación de instrucciones y facilita la optimización de traductores binarios.

Deflater funciona como una caja negra, es decir, que no requiere acceso al código interno del traductor sino que este observa solamente el resultado de la traducción.

La herramienta incluye un modelo matemático que calcula la inflación global, además de un simulador que permite observar el comportamiento del código traducido en tiempo real.

También cuenta con pruebas automatizadas que evalúan diferentes traductores sin necesidad de intervención manual.

Gracias a estas características Deflater ha mejorado el rendimiento de traductores como QEMU, logrando incrementos significativos tras aplicar sus recomendaciones.

Esto demuestra que, con un análisis detallado, es posible reducir la sobrecarga producida por la traducción entre arquitecturas, alcanzando una ejecución más eficiente en plataformas modernas.

<https://dl.acm.org/doi/pdf/10.1145/3640813>



