

Для современных микроархитектур критически важным фактором производительности является непрерывность подкачки кода и данных. Частота промахов в кэш инструкций названа существенным фактором, ограничивающим производительность типичных задач в датацентрах поисковых компаний [1].

В данной работе рассматривался эффект применения технологии двоичной трансляции с точки зрения оптимальности использования кэша инструкций (L1i) современных микропроцессоров. Исследование показало, что двоичный транслятор способен существенно улучшить использование кэша инструкций за счет специальной схемы размещения инструкций в памяти.

Основной метрикой, позволяющей сравнивать способы размещения инструкций, является число промахов в кэш на тысячу динамически исполненных инструкций (МРКИ, instruction cache misses per 1000 instructions).

Можно выделить два класса факторов, влияющих на МРКИ оттранслированного кода – имеющие положительный эффект (уменьшение числа промахов) и отрицательный эффект. К положительным факторам относятся оптимизации, устраняющие избыточность и уменьшающие размер кода (control- и data-flow оптимизации), а также фактор последовательной раскладки кода. При последовательной раскладке уменьшается вероятность попадания многих блоков памяти с инструкциями (cache line) в одно и то же множество ассоциативности (cache set) и тем самым повышается эффективность использования кэша. Негативными являются все преобразования, приводящие к увеличению объема кода: подстановка процедур, дублирование инструкций цикла, добавление новых инструкций (компенсирующий код).

Эффект от последовательной раскладки в наибольшей степени проявляется в случае циклов, имеющих некомпактное, разреженное размещение инструкций. Если блоки с инструкциями размещены таким образом, что они претендуют на одно множество ассоциативности, и число блоков превышает размер ассоциативности кэша, то происходит циклическое вытравливание инструкций из кэша, что приводит к задержке в исполнении программы. Для таких программ характерен шаблон использования множеств кэша, в котором легко различимы множества с избыточным давлением (рис. 1).

Результаты экспериментов, проведенные на наборе интервалов из тестового набора SPEC2006, показали значительный выигрыш в производительности между оригинальным и оптимизированным кодом. При этом среднее число промахов на тысячу инструкций (МПКИ) для оригинальных и оптимизированных кодов составляет 0.6 и 0.4, соответственно.

В рамках работы также была предложена схема компактной раскладки кода, нацеленная на добавленные компенсирующие инструкции, исполняемые в исключительных случаях. При раздельном размещении компенсирующих инструкций удастся еще улучшить картину использования кэша инструкций. Это дополнительно дает небольшое увеличение производительности относительно совместной раскладки (+0.1% в геометрическом среднем).

Литература

1. Kanev S., Darago JP, Hazelwood K., Ranganathan P. Profiling a warehouse-scale computer // Proceedings of the 42nd Annual International Symposium on Computer Architecture. ISCA '15. New York, NY, USA: ACM – 2015, P. 158—169

Рис. 1. Пример избыточного давления на кэш, вызванный некомпактностью циклического кода.

