

Реализация индексного анализа для деревьев циклов любого вида сложности

П.А. Юдин

Московский физико-технический институт (государственный университет)

yudin@frtk.ru

При исполнении большинства вычислительных задач основная часть времени тратится на вычисления, которые содержатся внутри циклов [1], поэтому автоматическое распараллеливание нацелено в первую очередь на работу с циклами. Для принятия решения о распараллеливании в компиляторе вычислительных комплексов «Эльбрус» проводится индексный анализ.

Исходный индексный анализ имеет ряд существенных недостатков, которые не позволяют применять его к циклам любой сложности. К примеру, если операций принадлежат одному охватывающему, но разным внутренним циклам, анализ не может выявить зависимость (независимости) операций.

Разрабатываемое улучшение анализа, применяемого сейчас в оптимизирующем компиляторе вычислительных комплексов «Эльбрус», позволит разрывать подобные зависимости, что существенно для применения техники автоматического распараллеливания к циклам со сложной структурой. Улучшенный алгоритм, используемый для разрыва зависимостей, рассматривает множество пар операций вида запись/запись и чтение/запись. Для каждой пары определяется зависит ли она от индуктивной переменной, при положительном результате проверяются операции на соответствие заданной канонической форме, задаются матрицы уравнений (с учетом границ циклов), решение которых ищется с помощью симплекс метода [2][3], а после решения определяется наличие зависимости или независимости операций.

Реализованный анализ позволил распараллелить циклы процедуры **zgemm**, которая принадлежит набору тестов **wupwise.168** [4] пакета тестирования **Spec2000**. В процессе произведенных улучшений на этом тесте удалось получить следующие результаты: время исполнения последовательной версии составило 762.35 сек., время исполнения исходной параллельной - 612.16 сек., время исполнения улучшенной параллельной – 526.17 сек. Таким образом, распараллеливание с использованием исходного анализа дает прирост производительности равный 20%, в то время как распараллеливание с использованием нового анализа - прирост в 31%.

Литература

1. *Muchnick Steven S. Advanced Compiler Design and Implementation. –1997.*

2. *George B. Dantzig and Mukund N. Thapa*. Linear programming 1: Introduction, Springer-Verlag. – 1997.
3. *Evar D. Nering and Albert W. Tucker*. Linear Programs and Related Problems. –Academic Press, –1993.
4. <http://www.spec.org/cpu2000/CFP2000/168.wupwise/docs/168.wupwise.html>