

Московский физико-технический институт
(государственный университет)
Физтех-школа радиотехники и компьютерных технологий
Кафедра радиоэлектроники и прикладной информатики

Адаптация оптимизации выделения маловероятных регионов цикла для программ с неточным профилем

Выпускная квалификационная работа
(бакалаврская работа)

Выполнил студент: Рябцев Д. И., 418 гр.

Научный руководитель: Линчик М. И.

Москва 2018

Профиль исполнения программы

Профиль – структура данных, хранящая статистику о частоте исполнения отдельных участков программы.

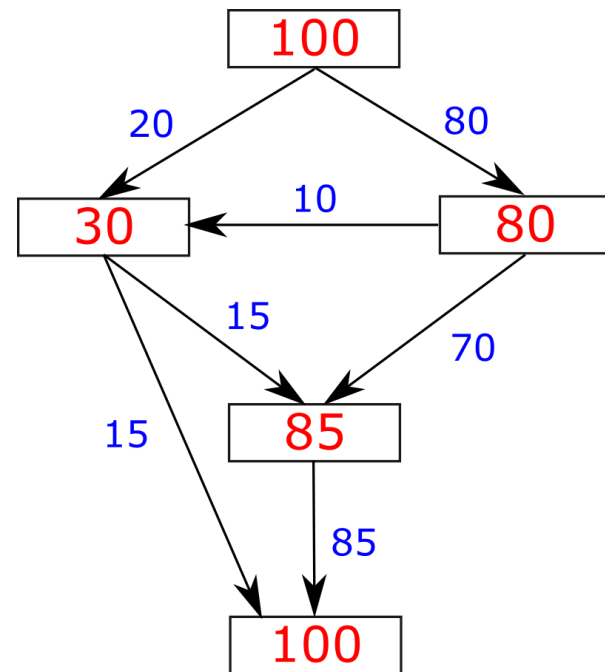
В компиляторе Эльбрус применяют два вида профиля:

- Предсказанный – формируется компилятором, основываясь на внутренних эвристиках;
- Реальный – полученный в результате запуска программы на тестовых данных с инструментированием кода.

Программа внутри компилятора представлена в виде графа потока управления (cfg-графа).

В cfg-графе профильная информация отражена в виде

- Счетчиков узлов (выделены **красным**)
- Счетчиков дуг (выделены **синим**)



Пример распределения счетчиков узлов и дуг на области cfg-графа

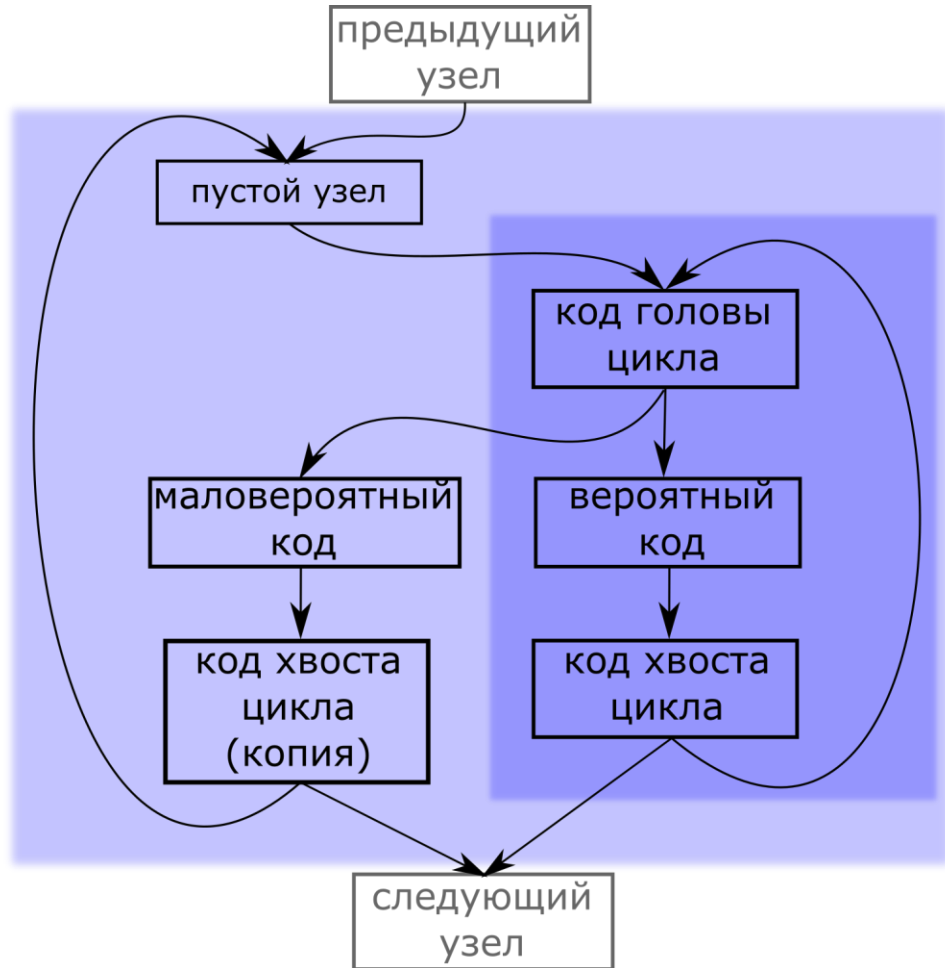
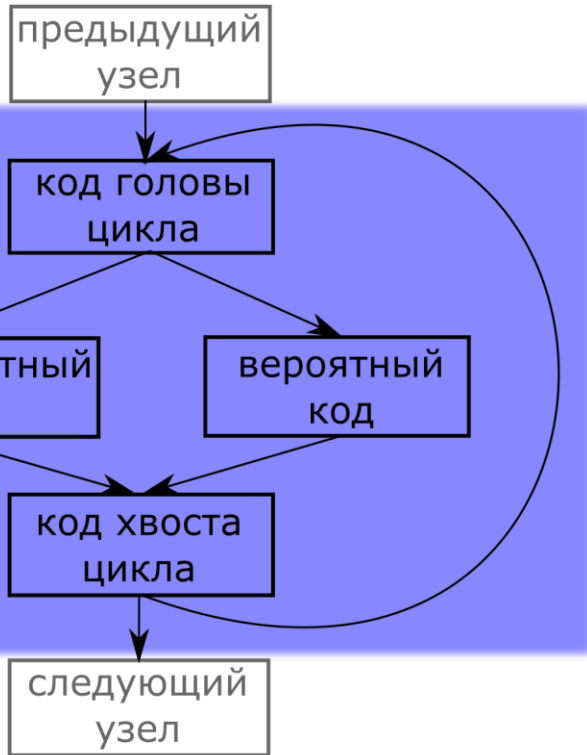
Введение в проблему

Фазы анализа многих оптимизаций используют профиль для принятия решение о необходимости применения оптимизации.

Предсказанный профиль бывает неточен, так как эвристики, на которых он основан, не во всех случаях способны правильно предсказать поведение программы.

В условиях неточного профиля фаза анализа может выработать неверное решение, что приведет к неоптимальной трансляции кода.

Оптимизация nesting

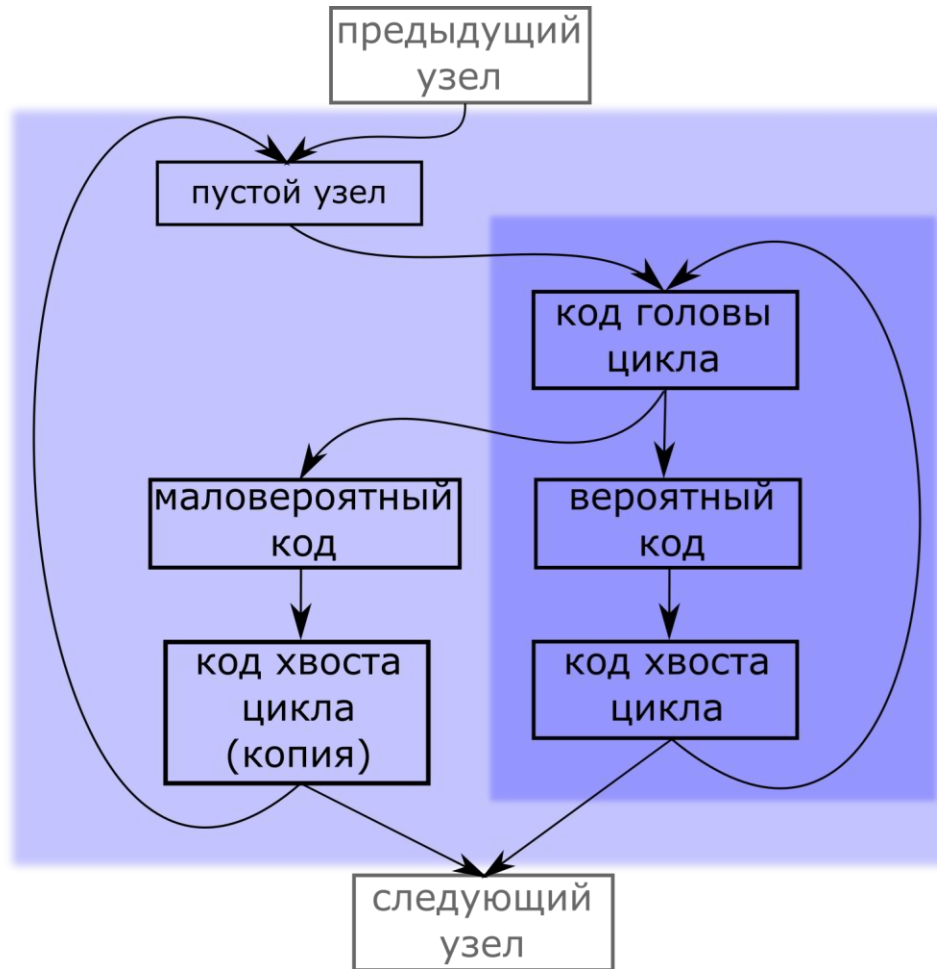


*цветом выделены узлы принадлежащие одному циклу, синий – изначальный цикл, голубой – созданный охватывающий

Алгоритм оптимизации nesting

Алгоритм:

- Фаза определения вероятной и маловероятной области цикла (используется профиль)
- Фаза преобразования nesting
 - формирование охватывающего цикла
 - вынос маловероятной области в охватывающий цикл
 - дублирование кода для устранения входов в результирующий цикл



Оптимизация способствует более оптимальному планированию и применению других оптимизаций.

Цель

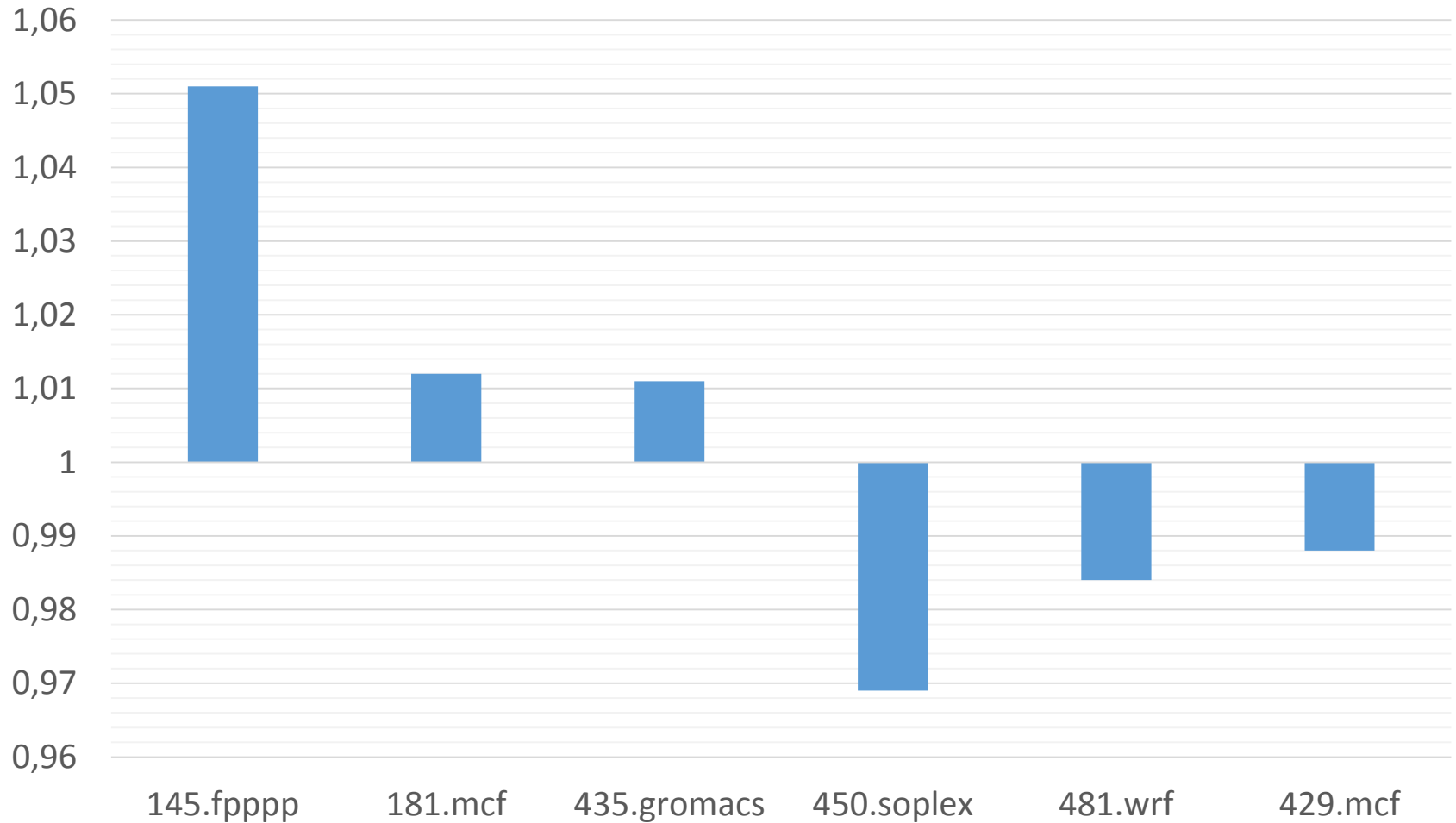
Адаптировать оптимизацию `nesting` для случая неточного предсказанного профиля

Задачи

- Исследовать эффективность применения оптимизации `nesting` при компиляции с использованием предсказанного профиля
- Изучить контекст задач, получивших ускорение от отключения оптимизации `nesting`
- Реализовать дополнительная фазу оптимизации, предназначенную для учета случая с неточным предсказанным профилем
- Усовершенствовать фазу определения вероятной области
- Произвести измерение производительности на пакетах SPEC.CPU 1995, 2000, 2006.

Исследование эффективности применения оптимизации nesting

Изменение производительности при отключении применения оптимизации nesting (SPEC CPU 1995, 2000, 2006)



*указаны результаты, превышающие 1%

Контекст задачи 145.fprrrr

Время исполнения задачи 145.fprrrr (указано в секундах)

С nesting	Без nesting
52.4	49.9

Время исполнения функции twldrv (указано в секундах)

С nesting	Без nesting
25.7	22.1

- Задача 145.fprrrr получила наибольшее ускорение (5.1%) за счет отключения оптимизации nesting
- Основной вклад в ускорение сделала функция twldrv
- По предсказанному профилю вероятность выйти в маловероятную область равна 0.04
- В ходе исполнения программы выходы в маловероятную область совершались с вероятностью 0.27
- Оптимизация nesting неэффективна из-за высокой вероятности выхода

Задача 145.fprrrr содержит контекст, иллюстрирующий уменьшение эффективности применения оптимизации nesting из-за неточности предсказанного профиля.

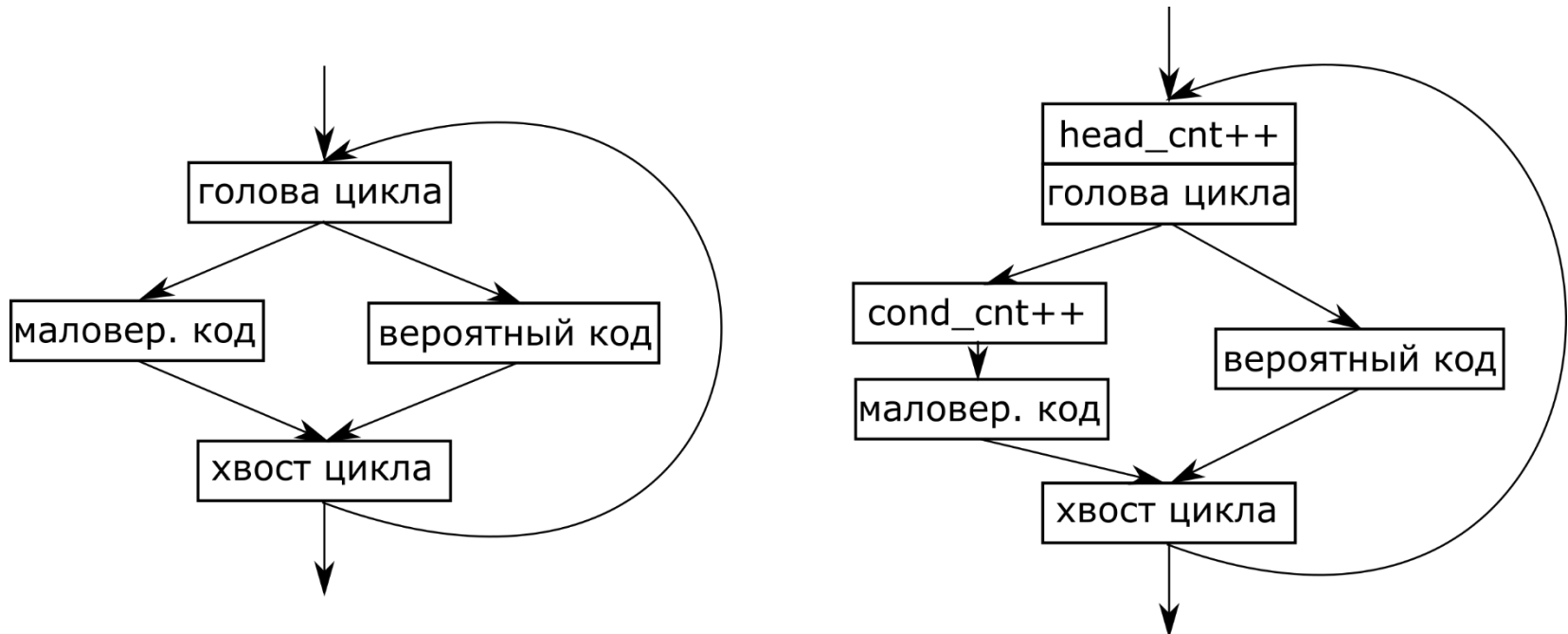
Была предложена схема динамического (происходящего во время исполнения) подсчета профиля.

Дополнительная фаза оптимизации для учета неточности предсказанного профиля

Схема динамического подсчета профиля

Вводим две статические переменные:

- `head_cnt` – подсчитывает общее число итераций
- `cond_cnt` – подсчитывает число выходов в маловероятную область

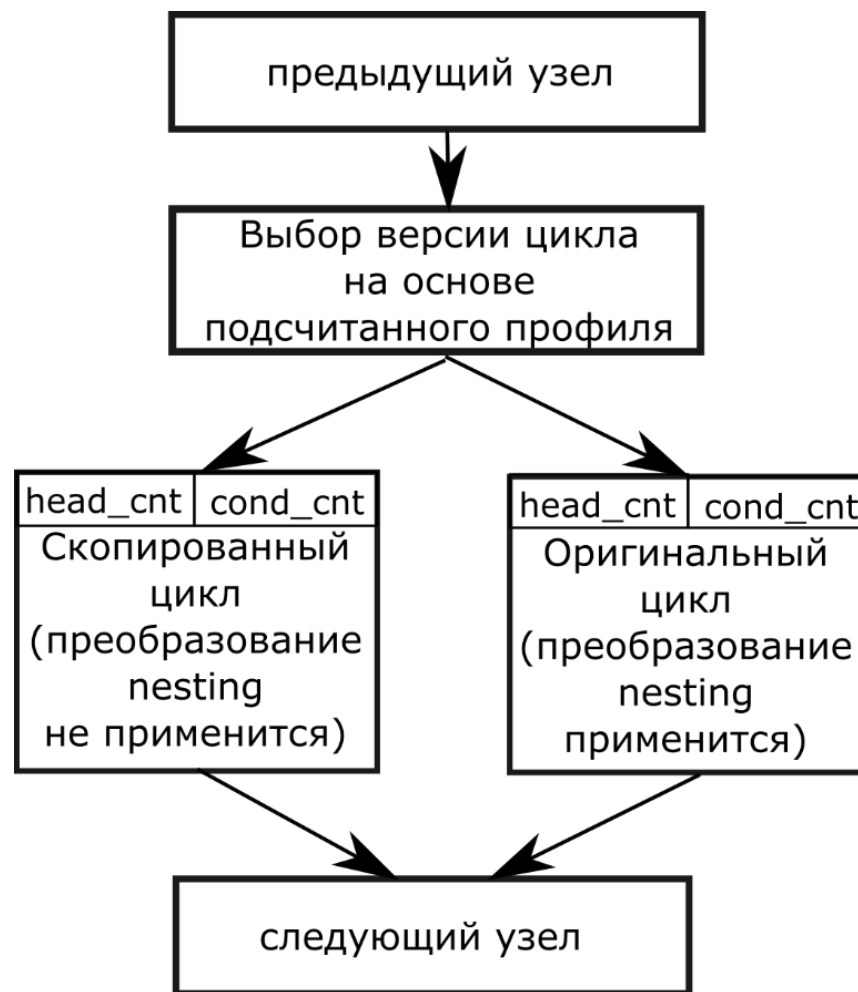


- Операции инкрементирования переменной `head_cnt` добавляем в голову цикла
- Узел с операциями инкрементирования переменной `cond_cnt` добавляем на дугу, ведущую в маловероятную область

Дополнительная фаза оптимизации для учета неточности предсказанного профиля

Динамическая проверка применимости оптимизации

- Исходный цикл дублируется, к оригиналу в дальнейшем применяется преобразование nesting, к копии – нет
- К обеим версиям цикла применяется схема динамического подсчета профиля
- Добавляется код, передающий управление на одну из версий цикла в зависимости от динамически подсчитанного профиля



Дополнительная фаза оптимизации для учета неточности предсказанного профиля

Выбор версии цикла на основе подсчитанного профиля

Условие передачи управления на оригинальный цикл:

$$\frac{cond_cnt}{head_cnt} < \left(\frac{1}{2}\right)^{shift}$$

Были проведены измерения производительности SPEC CPU 1995, 2000, 2006 при различных значениях параметра shift

shift	134. perl	147. vortex	200. sixtrack	176. gcc	181. mcf	186. crafty	454. calculix	462. libquantum	В среднем
2	-0.7%	-0.5%	-0.1%	-0.1%	+0.9%	+0.5%	0%	-3.2%	-0.1%
4	+1.1%	-0.5%	0%	+0.1%	+1.5%	-0.1%	0%	+2.1%	+0.1%
6	-0.8%	-1.5%	-2.7%	-3.1%	+2.5%	+1%	+1.1%	-1.1%	-0.2%

*приведены результаты, различающиеся больше чем на 1%

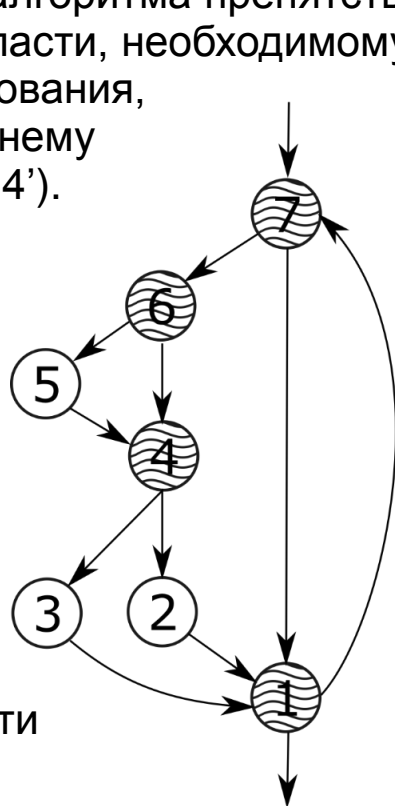
По результатам измерений было выбрано значение shift, равное 4

Фаза определения вероятной области цикла

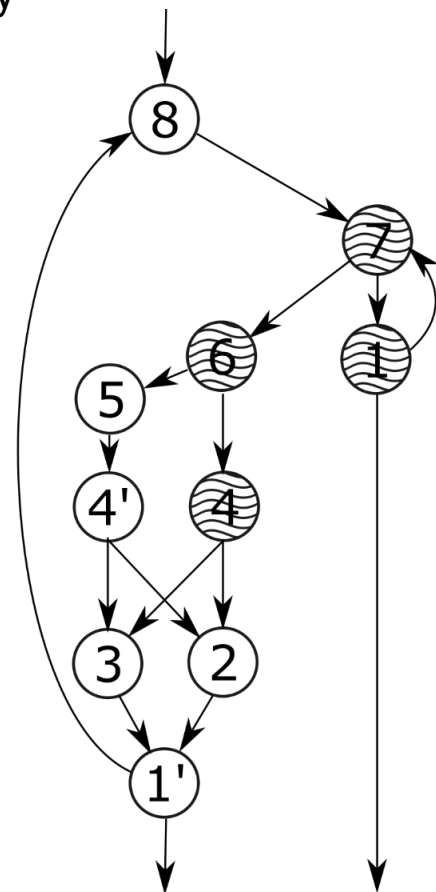
Обнаруженная проблема

Текущий алгоритм определения вероятной области цикла в некоторых случаях вносит в область узлы, которые в дальнейшем вынесутся из цикла (узлы 6 и 4).

Данное несовершенство алгоритма препятствует анализу выходов из вероятной области, необходимому для динамического профилирования, а также приводит к излишнему дублированию кода (узел 4').



исходный
цикл



результат
преобразования nesting



- узел вероятной области

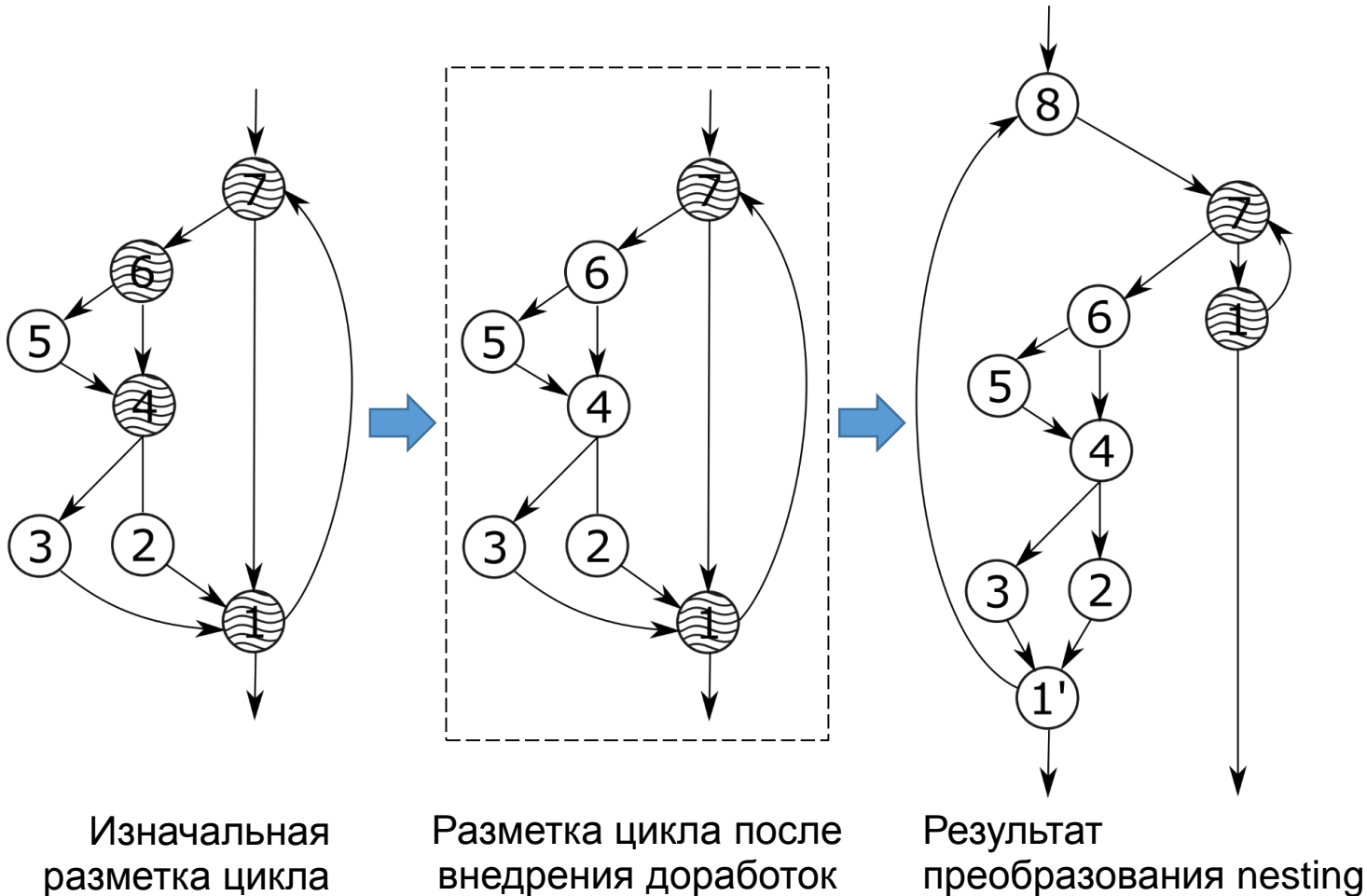


- узел маловероятной области

Фаза определения вероятной области цикла

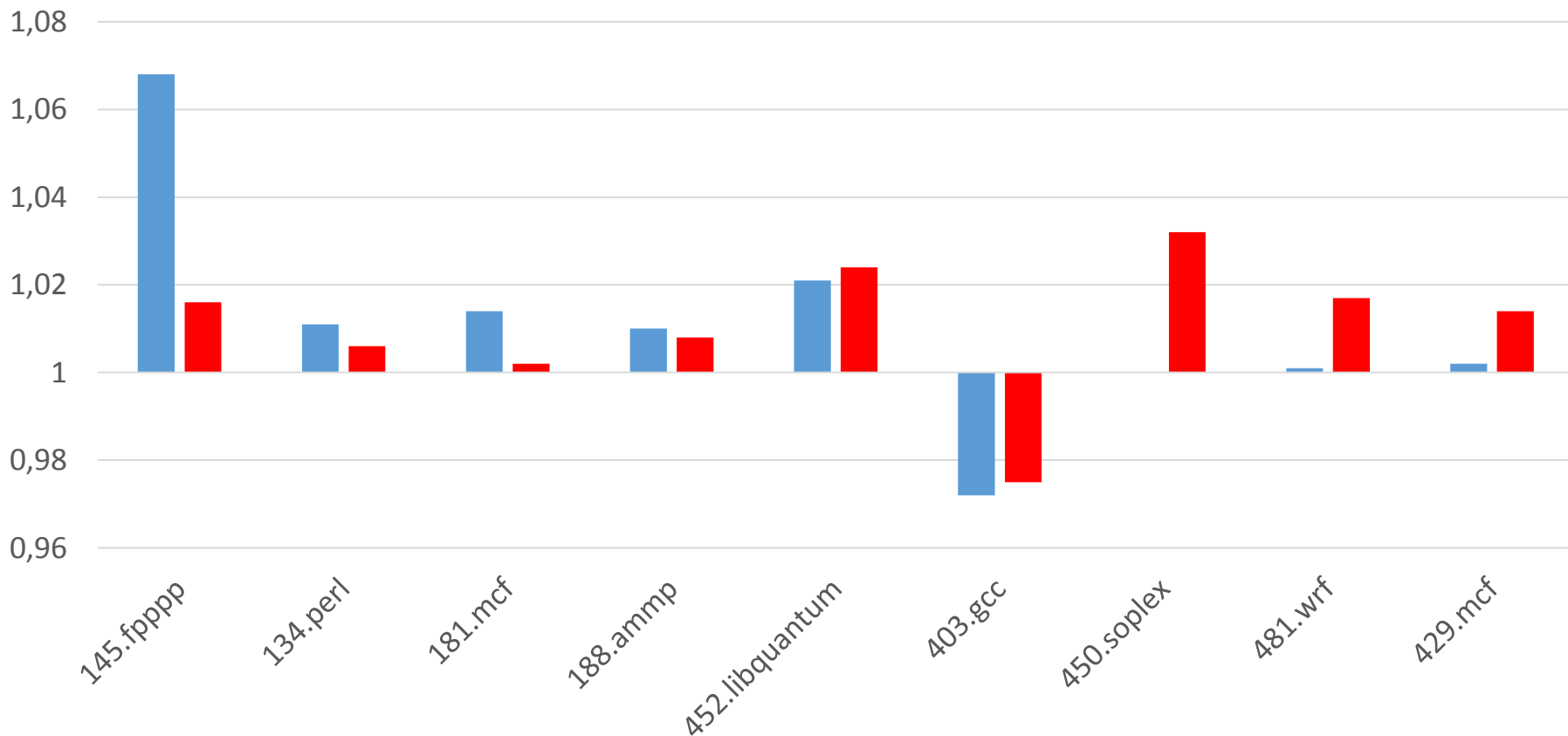
Реализованное решение

В алгоритм был добавлен дополнительный шаг, в ходе которого осуществляется обратный (postorder) обход в глубину узлов цикла, если узел не имеет потомков, принадлежащих вероятной области, он исключается из области.



Измерения производительности

Изменение производительности доработанной оптимизации по сравнению с исходной версией – **синие** столбцы, по сравнению с отключением оптимизации – **красные** столбцы (SPEC.CPU 1995, 2000, 2006)



*указаны результаты, превышающие 1%

- Максимальный прирост производительности составляет 6.8%
- В среднем все три пакета (72 задачи) ускорились на 0.1%

Результаты

- Исследована эффективность применения оптимизации `nesting` при компиляции с использованием предсказанного профиля
- Найден и изучен контекст задачи, в которой неточный предсказанный профиль приводит к ухудшению производительности
- Доработана оптимизация `nesting`
 - Реализована дополнительная фаза, предназначенная для учета случая с неточным предсказанным профилем
 - Усовершенствована фаза определения вероятной области цикла
- Произведены измерения производительности для внедренных доработок на пакетах SPEC.CPU 1995, 2000, 2006, показавшие прирост производительности до 6.8%.