

Московский физико-технический институт (государственный университет)
Факультет радиотехники и кибернетики
Кафедра информатики и вычислительной техники

Выпускная квалификационная работа (магистерская диссертация)

Оптимизация потоковой нагрузки на подсистему памяти методом управления режимом кэширования отдельных операций записи

Выполнил: Горелов Михаил, 913 группа

Научный руководитель: к.ф.-м.н. Нейман-заде М.И

Проблематика

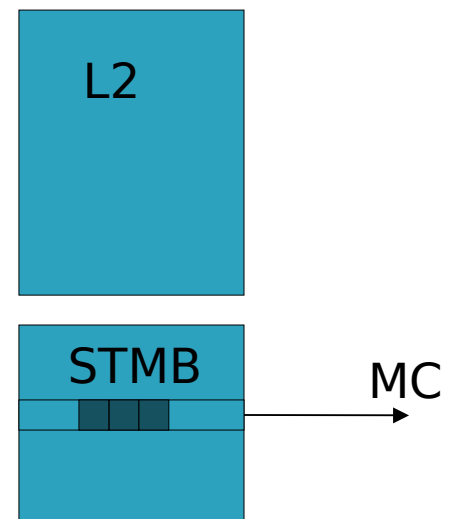
- ▶ Запись в цикле больших массивов может привести к ситуации, когда данные в кэш-памяти будут замещены раньше, чем будут использованы
- ▶ При промахе в кэш-память инициализируется запрос подложки в контроллер памяти и запись в память замещенной строки

Кэширование данных не всегда дает преимущество

Используемые решения

- ▶ Отключение кэширования
- ▶ Включение режима комбинированной записи (Write Combining)

- Данные не кэшируются
- Нет обращения в МС за подложкой
- Несколько записей по соседним адресам объединяются в одну
- Интервал между комбинируемыми записями по времени планирования не превышает 7 тактов



- ▶ **Режимы включаются вручную пользователем:**
 - с помощью разметки страниц
 - с помощью *built-in* функций

Постановка задачи

- ▶ Исследовать возможность автоматизации управления режимом кэширования отдельных операций записи в циклах средствами оптимизирующего компилятора Эльбрус
- ▶ Реализовать алгоритм оптимизации потоковой нагрузки на подсистему памяти, основанный на управлении режимом кэширования отдельных операций записи в циклах

Реализация

- ▶ Алгоритм применяется к гнёздам циклов



Анализ

Общий анализ гнезда циклов

- ▶ Проверяется структура гнезда
 - Наиболее часто встречаются гнёзда вида:

```
for (i1)
  for (i2)
    for (i3)
      {
        ...
      }
    }
  }
}

for (i1)
  for (i2)
    for (i3)
      {
        ...
      }
    }
  }
  for (i4)
    {
      ...
    }
  }
```

Анализ

Поиск подходящих операций записи

- ▶ Для каждого цикла находятся все операции записи и чтения
- ▶ Для каждой операции записи проверяется:
 - Линейность ее индекса относительно индуктивных переменных цикла;
 - Объект — массив;
 - Нет чтений по тому же адресу
(используется анализ зависимостей)
- ▶ Операции, записывающие в один массив, группируются

Анализ

Проверка времени планирования

- ▶ Строится предварительное планирование гнезда циклов
- ▶ Каждой операции каждого цикла ставится в соответствие время планирования в тактах
- ▶ Результаты работы планировщика:
 - Интервал между операциями
 - Время исполнения одной итерации цикла

Результатом работы аналитической части является модель работы с памятью - входные данные для оптимизации

Оптимизация

Вычисление размера записываемых данных

- ▶ *Критерий применимости: размер записываемых данных превышает размер L2*
- ▶ На этапе компиляции известны границы цикла => Статически
 - Вычисление размера записываемых данных и его сравнение с размером L2 происходит на этапе компиляции
- ▶ Границы неизвестны => Динамически
 - На этапе компиляции — изменение управляющего графа, построение арифметических операций
 - На этапе исполнения — вычисление размера записываемых данных и принятие решения о целесообразности применения оптимизации

Оптимизация

Вычисление размера записываемых данных

- ▶ Динамический алгоритм



Оптимизация

Разметка операций

- ▶ Для управления кэшируемостью операции используется спецификатор адреса памяти (mas)
- ▶ Интервал между записями по времени планирования не превышает 7 тактов
 - => Операции помечаются для применения к ним режима комбинированной записи
- ▶ Между операциями будет более 7 тактов
 - => Запрещается кэширование, режим комбинированной записи не включается

Проблемы, выявленные при практической реализации

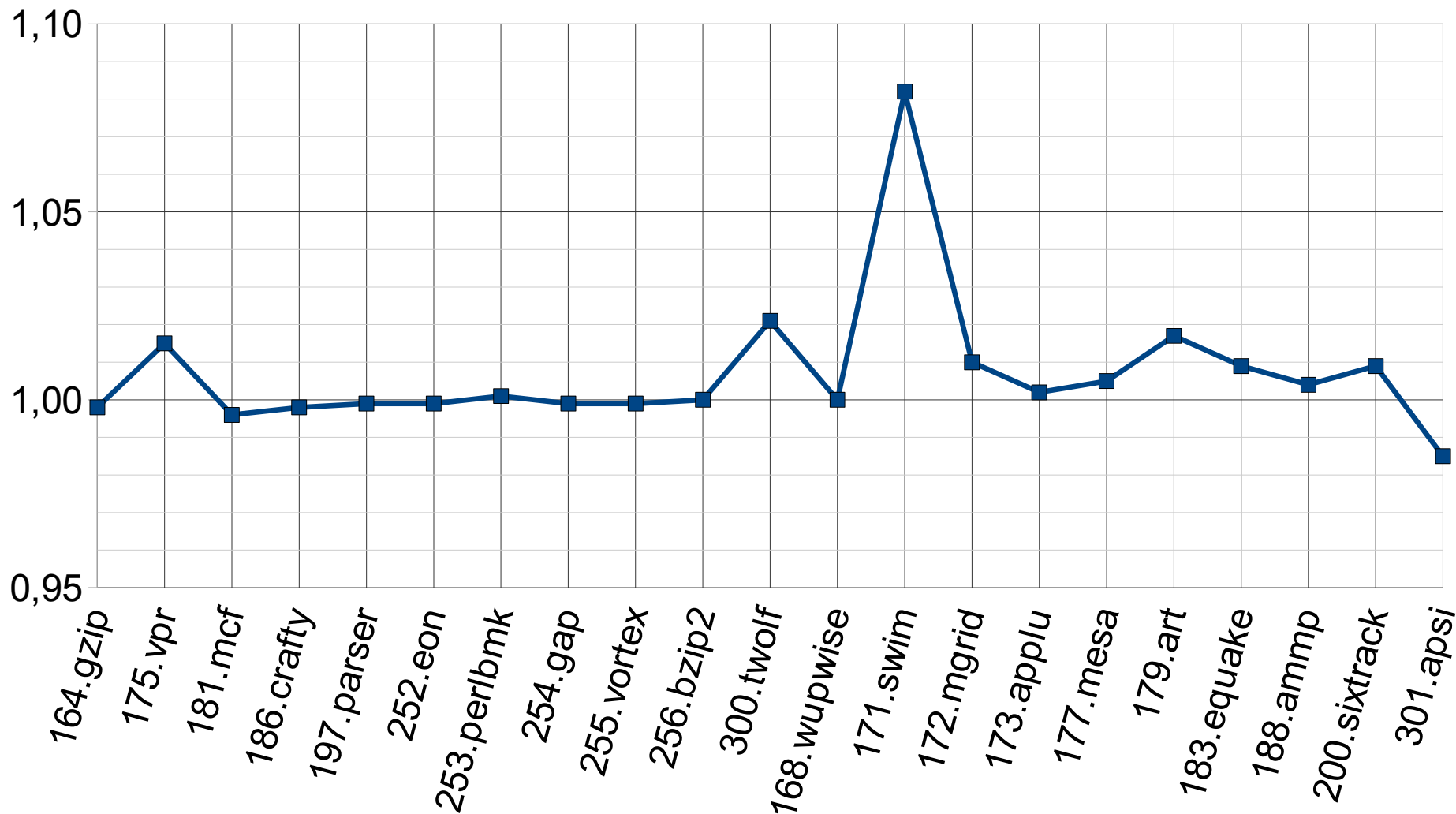
- ▶ Оптимизирующий компилятор не подготовлен к использованию режимов кэширования
 - Пример - задача 173.appli (Spec 2000):
 - простое гнездо циклов
 - 3 уровня вложенности
 - 100 подходящих операций записи
(запись в 4 разных массива)

На этапе финального планирования операции были перемешаны => расстояние между записями по соседним адресам стало более 7 тактов

Экспериментальные результаты

► Spec 2000

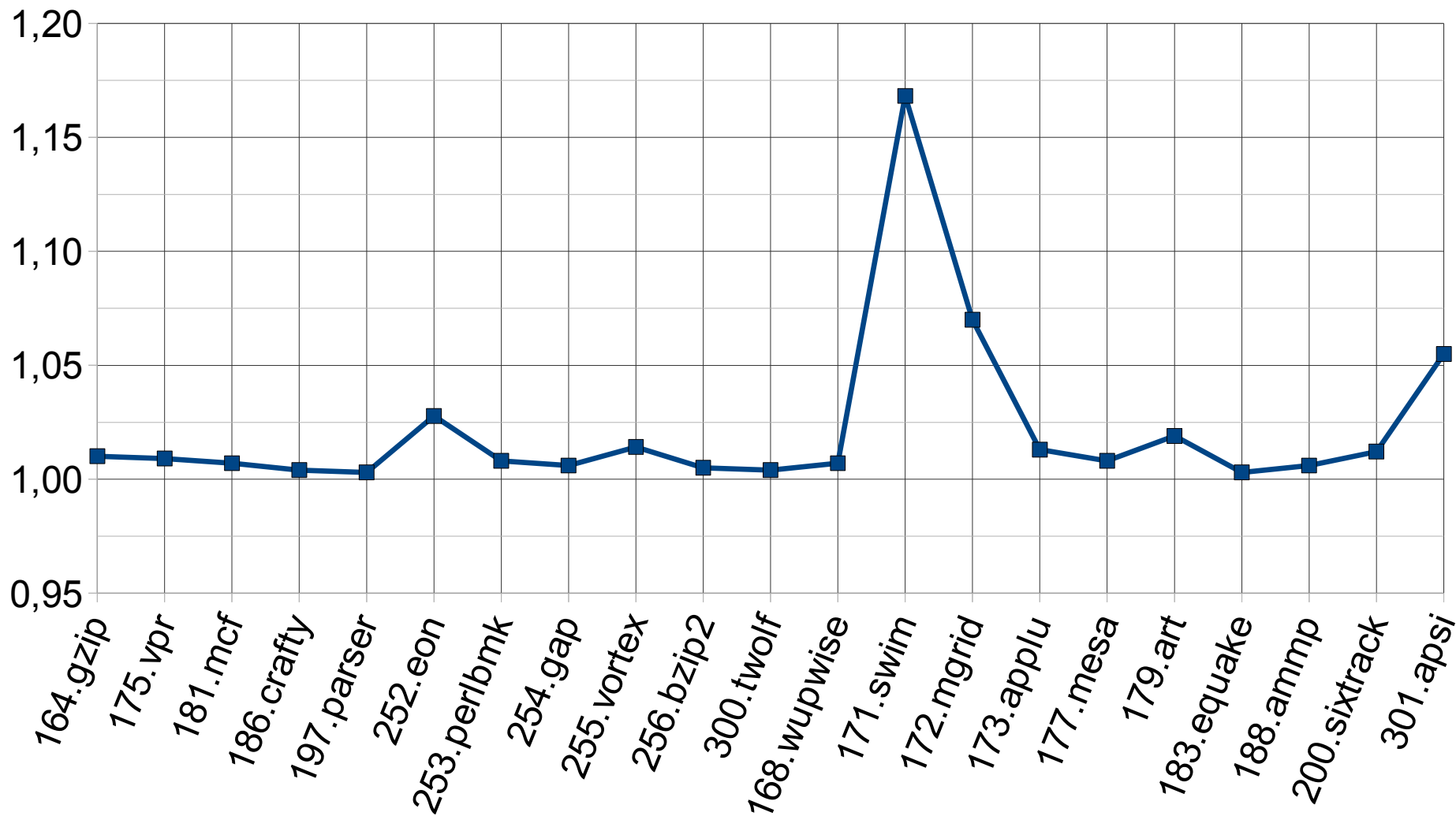
Увеличение производительности



Экспериментальные результаты

► Срес 2000

Изменение времени компиляции



Результаты работы

- ▶ Было проведено исследование возможности автоматизации управления режимом кэширования отдельных операций записи в циклах средствами оптимизирующего компилятора Эльбрус
 - ▶ На основании исследования был реализован алгоритм оптимизации потоковой нагрузки на подсистему памяти
 - ▶ Оптимизация была внедрена в оптимизирующий компилятор и протестирована на задачах пакета Spec2000
- 