Разработка средств автоматического подбора параметров параллельной программы на основе тестирования для оптимизации её нефункциональных свойств

Презентация подготовлена
Студентом 4го курса ФИТ
Чемагиным Александром Сергеевичем

Научный руководитель
Маркова Валентина Петровна

Соруководитель
Киреев Сергей Евгеньевич

Рецензент
Ракитский Антон Андреевич

Цель и задачи работы

Целью работы - разработка средства автоматизированного подбора параметров параллельной программы на основе тестирования для оптимизации её нефункциональных свойств.

Задачи:

- Выбрать методы для реализации.
- Разработать средство оптимизации параметров.
- Провести тестирование средства.

Требования:

- Процесс подбора параметров должен быть ограничен во времени.
- Взаимодействие с программой происходит методом чёрного ящика.

Особенности задачи

1. Функция оптимизируемой задачи может быть произвольной.
   1. Функция может иметь большое количество локальных оптимумов.
   2. Функция может иметь точки разрыва.
   3. Функция может быть не гладкой.
   4. И другое.

2. Есть возможность посчитать только значение функции, а не её производную.

3. Вычисление значения функции в точке - дорогостоящая операция.

Существующие средства подбора параметров

- Intel MPI — библиотека, реализующая интерфейс MPI
- Система автоматизированного распараллеливания Ratio
- Active Harmony — система подбора параметров во время исполнения
- …

Методы оптимизации

В рамках данной работы рассмотрено несколько методов оптимизации. А именно:

- Полный перебор
- Метод Монте-Карло или грубый случайный поиск
- Байесовская оптимизация
- Поиск по сетке
- Градиентный спуск
- Покоординатный спуск
- Генетические алгоритмы

Автоматизированное средство оптимизации. Ключевые возможности

1. Работа с разными типами параметров.
2. Возможность проводить оптимизацию разными методами.
3. Возможность ограничить время работы оптимизации.
4. Возможность проводить несколько замеров при одном наборе параметров.

Средство написано на языке Python.
Используемые библиотеки: Subprocess, Itertools и Bayesian Optimization.

Описание работы программы. Конфигурационный файл

```
{
    "program_path": "PATH/model_testrun",
    "save_path": "PATH/end/For_D.txt",
    "searching_word": "time",
    "find_type": 5,
    "find_condition": 30,
    "time_limit": 30000,
    "trails": 3,
    "data_set_len": 2,
    "data_set": [
        {
            "type": 1,
            "range": [
                50,
                1000
            ]
        },
        {
            "type": 3,
            "range": [
                10,
                20
            ]
        }
    ]
}
```

- Путь до программы или скрипта.
- Путь до файла, куда будут сохранены итоги всех запусков.
- Строка, которую программа будет искать в выводе программы.
- Номер алгоритма.
- Данные для алгоритмов.
- Ограничение по времени для оптимизации.
- Количество запусков при одних условиях.
- Количество параметров.
- Список параметров.

ОПИСАНИЕ РАБОТЫ ПРОГРАММЫ. ОПИСАНИЕ ПРОЦЕССА РАБОТЫ И АРХИТЕКТУРЫ

1. Модуль подготовки данных.
2. Модуль выбора значений для выбранного метода.
3. Модуль проведения запусков.
4. Модуль метода.
5. Модуль обработки конечных результатов работы.

ТЕСТ 1. ПЕРЕМНОЖЕНИЕ ДВУХ МАТРИЦ 1300*1300 MPI

Перемножение матриц 1300 на 1300 с использованием MPI.
Параметры: ключи компиляции, число процессов, два слабо влияющих параметра.
Цели:

1. Анализ ситуации, когда диапазоны значимых параметров малы.
2. Проверка влияние шума процесс оптимизации.

Алгоритм	Результаты	Время работы (секунды)	Вывод
Случайный поиск	(-O3, 4, 14, 1), (-O3, 5, 23, 78), (-O3, 5, 41, 84),(-O3, 4, 34, 2), (-O3, 4, 45, 67)	92, 207,133, 270, 164	Результат всегда верный
Байесовская оптимизация	(-O3, 4, 91, 53), (-O3, 4, 84, 72), (-O3, 4, 33, 51),(-O3, 4, 27, 89), (-O3, 4, 64, 5)	147, 132, 161, 138, 142	Результат всегда верный
Поиск по сетке (начальный шаг 30)	(-Os, 4, 42, 58)	1894	Результат неверный
Поиск по сетке (начальный шаг 20)	(-O3, 4, 78, 62)	2438	Результат верный

Тест 1. Итоги

- Байесовская оптимизация и случайный поиск всегда находили близкие к оптимальным параметры.
- В тесте с однократным запуском случайный поиск уступал байесовской оптимизации.
- Поиску по сетке не всегда удавалось найти близкие к оптимальным параметры.

Тест 2. Перемножение двух матриц 1300*1300 OpenMP

Перемножения двух матриц 1300 на 1300 блочным методом с использованием OpenMP.
Параметры: ключи компиляции, число процессов, три числа, отвечающих за размеры блоков.
Цели теста:
1. Проверить работу методов в случае больших диапазонов значений значимых параметров.
2. Оценить влияние большого количества параметров на работу методов.

Алгоритм	Найденные значения	Время работы(секунды)	Вывод
Поиск по сетке (начальный шаг 10)	(-Ofast, 7, 28, 28, 4)	8083	Результат верный
Поиск по сетке (начальный шаг 20)	(-O1 -march=native, 13, 17, 22, 4),	4277	Результат неверным
Поиск по сетке (начальный шаг 30)	(-O3 -march=native, 5, 30, 28, 4)	3684	Результат верный
Случайный поиск	(-O3 -march=native, 24, 29, 38, 7), (-Ofast, 35, 14, 23, 8), ...	290, 171, 135, 235, 163	Результат верный
Байесовская оптимизация	(-Ofast -march=native, 14, 35, 37, 8), (-O2 -march=native, 7, 38, 22, 4), ...	184, 176, 177, 189, 172	Результат верный

Тест 2. Итоги

- Байесовская оптимизация дала хорошие результаты во всех случаях.
- Случайный поиск не уступал байесовской оптимизации.
- Поиску по сетке не всегда удавалось найти близкие к оптимальным параметры.

Вывод из тестирования

1. Байесовская оптимизация показала себя лучше всего.
2. Случайный поиск хорошо показывает себя на большинстве задач.
3. Шум влияет на случайный поиск сильнее, чем на байесовскую оптимизацию.
4. Поиск по сетке не подходит для данной задачи.

Заключение

- Выполнен обзор и анализ методов, которые могут быть использованы в алгоритме поиска параметров.
- Разработан прототип средства подбора параметров для программ, взаимодействие с которыми происходит методом чёрного ящика.
- Выполнено тестирование разработанного средства. По результатам тестирования для поиска параметров рекомендуется использовать байесовскую оптимизацию.

СПАСИБО ЗА ВНИМАНИЕ!

N* Новосибирский государственный университет

*НАСТОЯЩАЯ НАУКА