

# ВИЗУАЛИЗАЦИЯ ТРАССЫ ПАРАЛЛЕЛЬНОЙ ПРОГРАММЫ В СИСТЕМЕ ПРОЕКТИРОВАНИЯ

Логинова И. П.

Объединённый институт проблем информатики Национальной академии наук Беларуси

Минск, Республика Беларусь

E-mail: irilog@mail.ru

*Разработан подход для визуализации и оценки результатов, полученных при проведении параллельных вычислений с использованием программ логической оптимизации для объектов, описания которых представлены в виде сетей независимых компонентов. Визуализация проводится на основе анализа пространственно-временных структур трассировочных метрик, полученных при выполнении параллельной программы.*

## ВВЕДЕНИЕ

В основе использования разработанного средства визуализации лежит выполнение параллельных процессов, в каждом из которых образован единственный поток [1]. В каждом процессе производится запуск команды, аргументы которой определяют имя программы и исходный объект. Если входные аргументы операции, выполняемой в данном потоке, не зависят от результатов операций, выполняемых с объектами в других потоках, то исходные объекты являются независимыми. Организовать совокупности независимых объектов для проведения параллельных вычислений с использованием программ оптимизации позволяет среда логического проектирования, в которой можно проводить ряд проектных операций для преобразования исходного объекта в одноранговую сеть независимых элементов. Таким образом, параллельные вычисления реализуются посредством запуска параллельных процессов, в каждом из которых выполняется программа для одного элемента сети.

В работе [1] представлены результаты экспериментов, которые свидетельствуют о эффективности параллельного запуска программ оптимизации на многоядерном процессоре. При проведении параллельного вычислений следует провести анализ распределения нагрузки по ядрам процессора, оценить время выполнения программы в каждом потоке. Все это важно, как для оценки эффективности такого подхода к оптимизации, так и для оценки быстродействия выполнения параллельной программы, в целом. С этой целью разработано программное средство, анализирующее трассу выполнения параллельной программы. Согласно данным этой трассы строится графическое отображение, которое дает наглядное представление о загрузке ядер процессора, о времени выполнения оптимизации для объектов с разными входными параметрами.

## I. ОРГАНИЗАЦИЯ ВИЗУАЛИЗАЦИИ ПРИ ВЫПОЛНЕНИИ ПАРАЛЛЕЛЬНЫХ ПРОЦЕССОВ

Средства визуализации используемые в параллельных вычислениях принято классифици-

ровать по таким категориям: системы визуального программирования (Visual Programming) и системы визуализации программирования (Visualization of Computing), средства визуального представления вводимой/выводимой информации. Если средства, относящиеся к первым двум категориям, призваны повлиять на процесс разработки параллельных программ, то использование визуальных средств, относящихся к последней категории, имеет практическое назначение, облегчает восприятие результатов работы параллельной программы и, как правило, предназначены для последующего анализа данных. Визуализация предстает наиболее естественным способом отображения пространственной модели параллельных вычислений: изменения происходят не только во времени, но и физически распределены в пространстве (различные процессы, выполняются на разных элементах вычислительной системы одновременно). Поэтому для визуализации при параллельных вычислениях используются двумерные, трехмерные модели.

## II. ОПРЕДЕЛЕНИЕ НАБОРА ХАРАКТЕРИСТИК, СОСТАВЛЯЮЩИХ ТРАССУ

Средства визуализации могут использовать самые различные методики сбора данных и связанный с ним способ организации работы визуального средства. Существует два различных подхода: первый и более простой в реализации off-line анализ, а также более прогрессивный и информативный on-line анализ. Off-line анализ подразумевает обработку данных после выполнения параллельной программы. Эта обработка основывается на том, что в процессе выполнения программы создается так называемая трасса - журнал, содержащий информацию о ходе выполнения программы. Сбор информации происходит в конкретных участках программы, которые определяются, как правило, при разработке параллельной программы. Обычно эти записи включают в себя временные отметки событий, происходящих в отдельных процессах, взаимодействие процессов между собой и другую информацию. После завершения программы файл трассы сохраняется. Современ-

ные средства визуализации на основе файла трасс позволяют строить различные схемы, иллюстрирующие состояние программы в каждый момент времени или, например, диаграммы, отображающие статистику работы параллельной программы. Проблемой сбора данных для визуализации является тот факт, что такие средства не могут не влиять на саму программу. В процессе off-line трассировки также происходит вмешательство в ход выполнения программы, но это воздействие неизмеримо меньше, чем возможное влияние on-line средств.

Для организации визуализации параллельных вычислений, в которых отсутствует взаимодействие между параллельными процессами, реализованы следующие этапы:

- трассировка параллельной программы с реализацией минимального воздействия на ход выполнения программы;
- анализ информации трассы;
- масштабируемая визуализация трассы.

Первый этап выполняется в программе, запускающей параллельные процессы. Трасса для одного процесса содержит значения, полученные в разных состояниях этого процесса:

- порядковый номер элемента сети;
- идентификатор параллельного процесса;
- время начала выполнения программы;
- время завершения программы в процессе;
- длительность выполнения программы.

Перед завершением работы параллельной программы все трассы процессов собирают вместе в один файл. Далее проводится анализ данных трассы осуществляемый уже в программе визуализации, где решаются следующие задачи:

1. Подготовка трассы для отображения в графическом визуализаторе;
2. реализация графического визуализатора на основе библиотеки QtCharts.

Основу программы визуализатора составляет подготовка трассы, заключающейся в организации многоуровневой сортировки хэш-таблицы, построенной по трассировочной информации. В качестве средства визуализации данных трассы, подготовленных на предыдущем этапе, используется модуль графической библиотеки QtCharts, который интегрирован в библиотеки Qt (начиная с версии Qt5.7). Библиотека является кроссплатформенной, достаточно легковесной, поддерживает широкий спектр разнообразных типов визуализаций (графиков, гистограмм, диаграмм и др.), обеспечивает высокую производительность. QtCharts предоставляет набор типов диаграмм, которые можно легко интегрировать в современные пользовательские интерфейсы. Для графического отображения выбран тип диаграммы в виде горизонтальной диаграммы с накоплением (HorizontalStackedBarChart). Этот тип диаграммы представляет данные трассы в виде горизонтально сложенных столбцов, сгруппированных по сле-

дующим категориям: по вертикали представлено распределение процессов по ядрам процессора; по горизонтали представлена шкала времени выполнения параллельной программы. В каждом горизонтальном столбце последовательное сложение блоков разной длины определяет порядок выполнения процессов, а размер блоков определяется длительностью работы программы в процессе, которая определяется порядком временных событий в каждом процессе. Примеры горизонтальной столбцовой диаграммы с накоплением представлены на рис. 1 и 2.



Рис. 1 – 16 ядер; 1 проект; 15 независимых элементов; программа ABC

На рис.1 представлена диаграмма, где отображено выполнение в параллельном режиме оптимизации (программа оптимизации ABC) для объекта проектирования, представленного в виде двухуровневой сети из 15-ти элементов. Эффективность использования параллельных вычислений, загруженность ядер и другие оценки определяются даже с первого взгляда. Следует отметить, что в параллельных процессах могут выполняться несколько программ оптимизации, выполняемых для нескольких исходных объектов. На рис. 2 представлена диаграмма, где отображено результаты выполнения в параллельном режиме оптимизации (программа ABC) для 30-ти элементов, представленных двухуровневыми описаниями двух проектов.

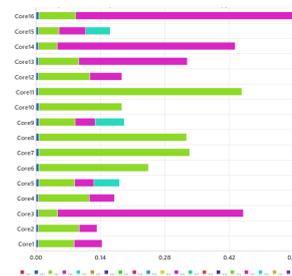


Рис. 2 – 16 ядер; 2 проекта; 30 независимых элементов; программа ABC

1. Логинова, И. П. Анализ быстродействия при реализации параллельных вычислений для решения задач оптимизации в системе логического проектирования /И. П. Логинова // BIG DATA и анализ высокого уровня: сб.науч. ст. IX Междунар. науч.-практ. конф. В 2 ч. Ч. 1. (Республика Беларусь, Минск, 17–18 мая 2023 года). – Минск: БГУИР, 2023. – С. 365-378.