

Анализ предметной области и существующих работ по тематике курсового проекта

1. Предметная область проекта

Целью данной работы является разработка программы, направленной на симуляцию физической модели и изучение эффективности применения методов параллельного программирования к ней.

Параллельное программирование – это подход к разработке программ, в котором задачи выполняются на нескольких процессорах или ядрах процессора. Вместо того, чтобы выполнять код последовательно, подход позволяет распределить вычислительную нагрузку между несколькими вычислительными ресурсами, такими как процессоры, для более эффективного выполнения. Параллельное программирование позволяетратно уменьшить время, затраченное на выполнение поставленной задачи.

Для исследования эффективности применения подхода обычно используются два показателя: эффективность распараллеливания и ускорение распараллеливания. Первый показывает, насколько хорошо применен подход, второй – как количество выделенных ресурсов влияет на эффективность [7].

2. Анализ аналогичных проектов и существующих решений для реализации проекта

В Работы по теме параллельного программирования в основном связаны с разделением основную задачу программы на несколько подзадач, которые выполняются параллельно друг другу, что в итоге приводит к более быстрому получению результатов.

Большинство проектов с открытым исходным кодом используют общедоступные библиотеки для реализации параллельности в программном продукте, например OpenMP [1] или OpenMPI [2] из-за их простой интеграции

в проект (OpenMP поддерживается в компиляторе GCC «из коробки»). В более серьезных проектах используются такие библиотеки как OpenCL или thread для получения доступа к аппаратным ресурсам на низком уровне.

В отдельных случаях разработчик проекта может применить GPGPU вычисления, что подразумевает использование ресурсов графического чипа и видеопамяти для решения задачи. Реализация такого подхода очень трудоемка, так как фактически в персональном настольном компьютере GPU является отдельным устройством и не имеет доступа к памяти программы, а программа не может напрямую взаимодействовать с ресурсами графического чипа.

В основном расчет физической модели используется в компьютерной графике для видео игр и кино, например для просчета объемного тумана или создания динамически изменяющихся объектов окружения. Существует несколько реализаций аналогичных проектов с использованием параллельного программирования, например встроенный в программный пакет Unity [3], предназначенный для разработки видео игр, модуль «particle system». Этот модуль реализован достаточно эффективно, чтобы работать в реальном времени и при этом не перегружать систему. Точных расчетов эффективности реализации «particle system» получить не представляется возможным, так как Unity [3] автоматически определяет, сколько ресурсов выделять как для отдельного кадра, так и для уникальной конфигурации системы пользователя.

Второй пример применения представлен в профильном программном пакете «houdini fx» [5], предназначенном для создания реалистичных объёмных или двумерных компьютерных эффектов. В «houdini fx» используются OpenCL [6] и OpenMP[1] с точной настройкой процесса просчета, программный пакет предоставляет возможность использования нескольких процессоров, видеокарт и специализированных ускорителей.

Основным методом оценки успеха в распараллеливании программы являются две метрики: ускорение и эффективность ускорения. Ускорение

вычисляется делением времени выполнения непараллельной программы на время выполнения параллельной программы. Под эффективностью распараллеливания понимается отношение получаемого ускорения к числу задействованных процессов [4].

3. Заключение

В результате обзора литературы был выявлен набор инструментальных средств для проведения исследования эффективности распараллеливания задач расчета и отображения симуляции физической модели.

Однопоточный вариант проекта будет реализован на языке программирования C++, с использованием системы сборки Cmake и компилятором GCC.

Наиболее подходящим вариантом для реализации проекта с поддержкой параллельных вычислений является библиотека OpenMP для распараллеливания на CPU.

Литература

1. Документация OpenMP [Электронный ресурс]:
<https://homepages.math.uic.edu/~jan/mcs572f16/mcs572notes/lec09.html>
2. Документация OpenMP [Электронный ресурс]:
<https://www.open-mpi.org/doc/>
3. Документация Unity [Электронный ресурс]:
<https://docs.unity3d.com/Manual/index.html>
4. Тютюнник, М. Б. Разработка и исследование продукционной системы параллельного программирования : специальность 05.13.11 "Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей" : автореферат диссертации на соискание ученой степени кандидата технических наук / Тютюнник Михаил Борисович. – Владивосток, 2010. – 19 с. – EDN QHHYAP.
5. Документация Houdini fx [Электронный ресурс]:

<https://www.sidefx.com/docs/houdini/>

6. Документация OpenCL [Электронный ресурс]:

<https://registry.khronos.org/OpenCL/sdk/3.0/docs/man/html/>

7. Michael J. Quinn. 1994. Parallel computing (2nd ed.): theory and practice. McGraw-Hill, Inc., USA.