

Модель программирования параллельных вычислений GraphPlus: примеры использования

С.В. Востокин

Представлены примеры реализации типовых схем параллельных вычислений master-worker с pull-стратегией и распараллеливания сеточного метода с линейной декомпозицией данных на языке C++ с библиотекой времени исполнения модели GraphPlus для платформы Win32. Рассмотрены преимущества предлагаемой архитектуры параллельных программ.

Архитектура современных процессоров такова, что прикладная программа полностью использует возможности аппаратуры, если в ней в явном виде реализована поддержка параллельных вычислений. Только правильно спроектированные многопоточные программы могут эффективно исполняться на многоядерных процессорах. Однако программирование на уровне модели потоков операционной системы сложно и чревато появлением трудно обнаруживаемых ошибок.

Проблема сложности параллельного программирования аналогична проблеме 60х годов прошлого века, проявившейся при написании больших программ. В программной инженерии известно, что решение заключается в систематическом следовании определенным правилам написания кода и развитии методов его повторного использования. Выражением данных правил является модель программирования и средства ее реализации.

В настоящее время в параллельном программировании используются средства MPI и OpenMP. Их недостаток - ориентация на процедурный стиль и не соответствие современной практике программирования. Мы считаем, что в области параллельного программирования объектные модели имеют преимущества перед процедурными, так как позволяют более удобно представить декомпозицию данных, параллелизм и взаимодействие.

Для описания семантики объектных моделей достаточно выразительных возможностей современных объектно-ориентированных языков C++, Java, C#, а некоторые аспекты модели удобно представить графически в виде диаграмм.

В качестве иллюстрации данных принципов на примерах простейших программ представляется реализация объектной визуальной модели параллельных вычислений GraphPlus (официальный сайт проекта: <http://graphplus.ssau.ru>). В основе механизма исполнения модели лежит специально разработанный паттерн «Постоялец-Посетитель», являющийся вариантом паттерна «Активный объект». Описаны приемы построения повторно используемых типовых схем параллельных вычислений master-worker с pull-стратегией и распараллеливания сеточного метода с линейной декомпозицией данных на языке C++. Показано, что для прикладного программиста кодирование состоит из порождения пользовательских классов из классов модели, создания их экземпляров и построения сетевых структур из прикладных и управляющих объектов модели. Синхронизация и взаимодействие передается связями в сетевой структуре, а их многопоточная реализация скрыта в механизме исполнения.

Рассматриваются преимущества предлагаемой архитектуры программ на основе модели GraphPlus. Модель единообразно реализуется на общеупотребительных языках ООП: C++, Java, C#. Программы с мелко и среднегранулярным параллелизмом, построенные в модели GraphPlus, эффективно выполняются. Это связано с тем, что предлагаемая реализация паттерна «Активный объект» позволяет избежать работы с динамической памятью, копирования блоков памяти, использует манипуляции с указателями, методы синхронизации в режиме пользователя, а не в режиме ядра. Модель ориентирована на построение библиотек типовых процессов, что способствует повышению уровня программирования. Модель допускает визуализацию, причем используются тривиальные методы построения кода по визуальному представлению. Это позволяет пользователю легко устанавливать соответствие между диаграммой и кодом при отладке.

Дальнейшие направления исследований по модели GraphPlus связаны с ее применением в адаптивных вычислениях в распределенных средах с флуктуирующими характеристиками.