ЕСТЕСТВЕННОНАУЧНОЕ НАПРАВЛЕНИЕ

РАЗРАБОТКА МАСШТАБИРУЕМЫХ МЕТОДОВ И ПАРАЛЛЕЛЬНЫХ АЛГОРИТМОВ ДЛЯ ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ДАННЫХ НА МНОГОПРОЦЕССОРНЫХ СИСТЕМАХ С ОБЩЕЙ И РАСПРЕДЕЛЕННОЙ ПАМЯТЬЮ

В настоящее время одним из феноменов, оказывающих существенное влияние на область технологий обработки данных, являются Большие Данные (BigData). В условиях современного информационного общества имеется широкий спектр приложений (социальные сети, электронные библиотеки, геоинформационные системы и др.), в каждом из которых производятся данные, имеющие сверхбольшие объемы и высокую скорость прироста (от 1 Терабайта в день). Интеллектуальный анализ предполагает поиск трендов и аномалий, скрытых в этих данных. В проекте исследуются методы и алгоритмы, обеспечивающие эффективный интеллектуальный анализ сверхбольших объемов данных на современных многопроцессорных системах с общей и распределенной памятью.

Руководители проекта - д.т.н. Л.Б. Соколинский, к.ф.-м.н. М.Л. Цымблер

ЦЕЛЬ РАБОТЫ

Разработка и исследование новых масштабируемых методов и параллельных алгоритмов интеллектуального анализа данных на гибридных многопроцессорных системах с многоядерными ускорителями класса MIC

ПУБЛИКАЦИИ

1 кандидатская диссертация

25 научных статей

17 научных докладов

ИНДЕКСИРОВАНИЕ

2 статьи в SCOPUS

23 статьи в РИНЦ

Сверхбольшие реляционные базы данных сохраняют в структурированном виде результаты интеллектуального анализа Больших Данных, требующие параллельной обработки. Существующие сегодня коммерческие СУБД, использующие фрагментный параллелизм (Teradata, Greenplum и др.), имеют высокую стоимость и ориентированы на специфические аппаратно-программные платформы. Альтернативой коммерческим СУБД являются свободные СУБД с открытым исходным кодом (PostreSQL, MySQL и др.).

Авторами предложена идея модернизации существующего исходного кода свободной последовательной СУБД для построения на ее основе параллельной СУБД для кластерных вычислительных систем путем внедрения фрагментного параллелизма.

Эффективное решение задачи разбиения графов имеет большое значение в ряде теоретических и практических задач: определения числа и состава компонент связности графа и представления графа в виде ярусно-параллельной формы, проектирование БИС (больших интегральных схем) и ПЛИС (программируемых логических интегральных схем), проектирование топологии локальных сетей, конечно-элементное моделирование и др.

Существующие алгоритмы предполагают возможность размещения графов и промежуточных данных обработки в оперативной памяти и неприменимы для случая сверхбольших графов. Был предложен метод обработки сверхбольших графов на основе использования параллельной реляционной СУБД PargreSQL. Метод предполагает представление графа в виде реляционной таблицы (списка ребер), которая распределяется по узлам кластерной системы и обрабатывается параллельной СУБД с помощью запросов SQL.

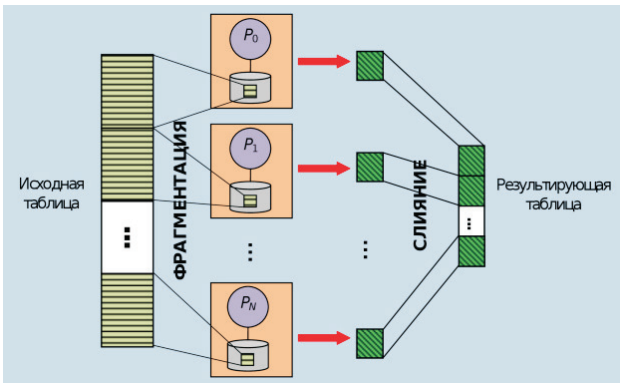
Задача поиска похожих подпоследовательностей возникает в широком спектре предметных областей: медицина, прогноз погоды, анализ движений, финансы и др. В настоящее время наиболее популярной мерой схожести подпоследовательностей является динамическая трансформация шкалы времени (DynamicTimeWarping, DTW), однако, несмотря на существующие техники ускорения, DTW остается вычислительно сложной операцией. Был предложен параллельный алгоритм поиска похожих подпоследовательностей временного ряда, совместно использующий мощности центрального процессора и многоядерного сопроцессора Intel Xeon Phi.

РЕЗУЛЬТАТЫ ПРОЕКТА

1. Разработана технология внедрения фрагментного параллелизма в свободные реляционные СУБД с открытым исходным кодом. Технология позволяет получить эффективное и относительно недорогое решение для организации хранения и обработки сверхбольших баз данных, обладающее хорошей масштабируемостью. На основе данной технологии разработана параллельная СУБД PargreSQL, являющаяся параллельной версией СУБД PostgreSQL (рис. 1).

2. Разработан метод разбиения сверхбольших графов на основе использования параллельной СУБД. Метод позволяет выполнять разбиение графов из миллионов вершин и ребер, которые не могут быть размещены в оперативной памяти (рис. 2).

3. Разработан параллельный алгоритм поиска похожих подпоследовательностей временного ряда для многоядерных сопроцессоров IntelXeonPhi. Алгоритм позволяет получить практически 100% использование сопроцессора в вычислениях и дает 10-ти кратное превосходство в быстродействии по сравнению с последовательным алгоритмом и 3-х кратное превосходство по сравнению с параллельным алгоритмом, не использующим сопроцессор (рис. 3).



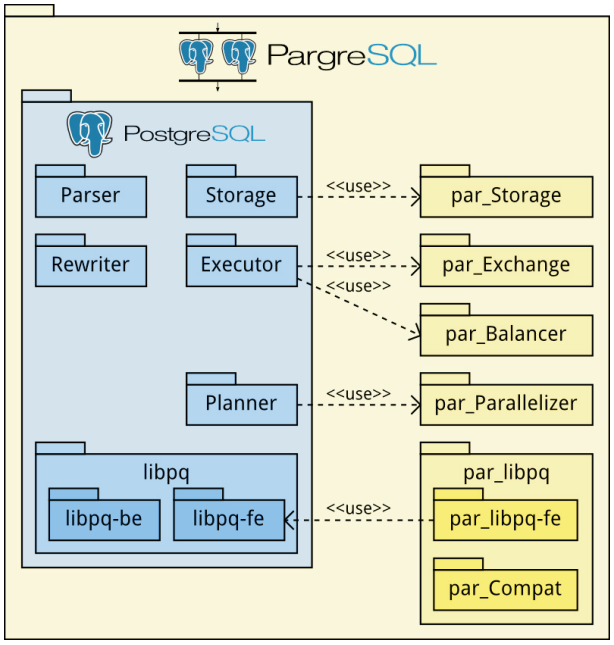


Рис. 1. Архитектура параллельной СУБД PargreSQL

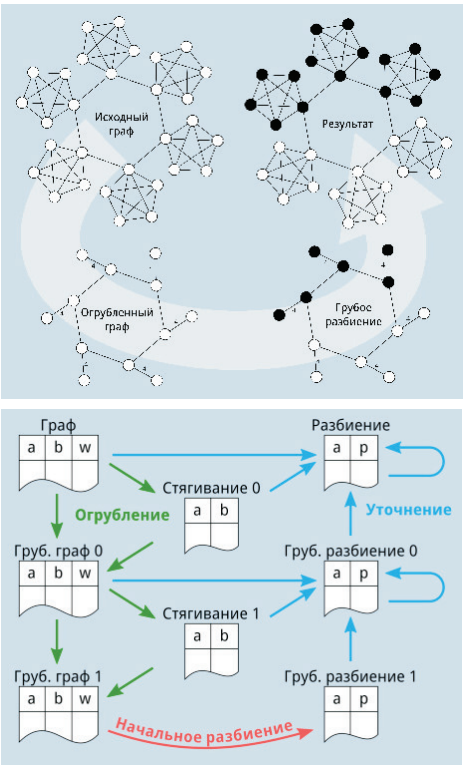
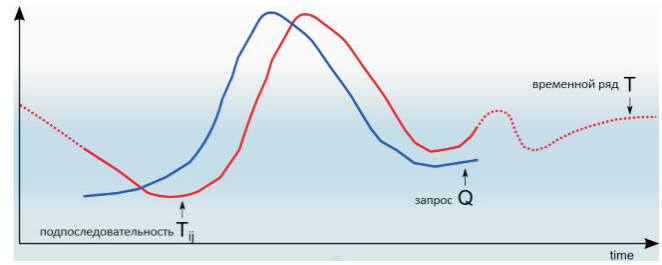


Рис. 2. Разбиение сверхбольших графов



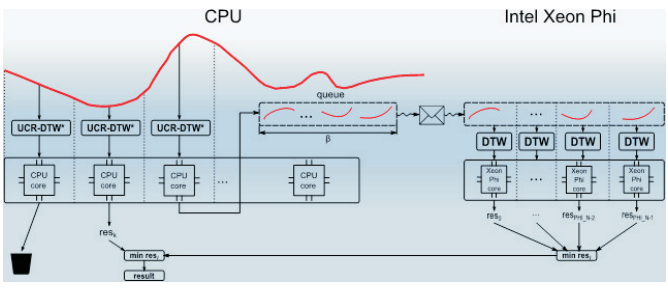


Рис. 3. Параллельный алгоритм поиска похожих подпоследовательностей временного ряда