ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

В.В. Фомин, Е.В. Петров

ОРГАНИЗАЦИЯ ПАРАЛЛЕЛЬНОЙ ОБРАБОТКИ ПРИ РЕАЛИЗАЦИИ WEB-СИСТЕМЫ ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ДАННЫХ

V.V. Fomin, E.V. Petrov

ORGANIZATION OF PARALLEL PROCESSING FOR IMPLEMENTATION OF WEB DATA MINING SYSTEM

Рассматривается проблематика обработки больших объемов данных при организации систем интеллектуального анализа, предлагаются решения повышения общей эффективности вычислительного процесса. Акцент делается на особенностях Grid-технологий, распределенных систем и параллельных вычислений в условиях Internet-сетей и гибкости подключаемых аппаратных ресурсов.

Ключевые слова: интеллектуальный анализ данных, Internet-сети, облачные вычисления, параллельные вычисления.

Focuses on the problems of processing large amounts of data in the organization of systems of mining, offers solutions to improve the overall efficiency of the computational process. The focus is on the features of Grid-technologies, distributed systems, and parallel computing in Internet-connected networks and flexibility of hardware resources.

Key words: data mining, Internet-networking, cloud computing, parallel computing.

Современные геоинформационные системы (ГИС) включают в себя возможности аналитических средств и кроме классических задач управления, контроля, статистического анализа и графической визуализации данных решают различные задачи поддержки принятия решений, прогнозирования, мониторинга. Области применения ГИС многочисленны и разнообразны, в том числе:

- картографии — учет и планирование географических, транспортных, экологических, экономических факторов;
- геология — поиск полезных ископаемых, сейсмо-прогнозирование и т.д.;
- сельское хозяйство — прогнозирование урожая, борьбы с вредителями и т.д.;
- метеорологии — исследование климата и факторов на него влияющих, выявление аномалий и предупреждение чрезвычайных ситуаций;
- административное управление — составление расписаний, оптимизация информационных потоков, мониторинг и контроль показателей деятельности;
- обороне и многих других областях.

По своему предназначению, ГИСы относятся к классу информационных систем обрабатывающих огромные объемы данных [1]. Развитие и применение методов интеллектуального анализа данных (распознавания, прогнозирования, машинного обучения и т.д.) выводят проблему обработки данных на приоритетный уровень научных и прикладных исследований.

За последние десятилетия произошло интенсивное развитие информационных систем [2], в том числе в таких областях как сетевые технологии Internet, способы хранения и представления знаний, языки и инструментарий программирования, методы искусственного интеллекта, алгоритмы распределенных и облачных вычислений и т.д.

Научно-технические достижения в области искусственного интеллекта повлияли на формирование новых и трансформацию старых классов информационных систем — интеллектуальные информационные системы, систем интеллектуального анализа данных, экспертные системы, системы поддержки принятия решений и пр.

С ростом качества и количества информации хранимой и обрабатываемой в современных информационных системах возрастает потребность в различного вида ресурсах (вычислительных, коммуникационных, информационных). Решение проблемы ресурсов осуществляется в рамках различных составляющих, в том числе:

1. Неуклонно увеличивающийся объем информации постоянно требует совершенствования и увеличения аппаратно-технических ресурсов-накопителей, процессоров, средств коммуникации и передачи информации и т.д. С позиции проектирования и эксплуатации информационных систем такой подход соответствует интенсивному развитию, а все инновации и экстенсивный потенциал отдаются на откуп развития вычислительной техники.
2. Появление новых требований к информационной парадигме компьютерных систем от хранения и поиска данных к представлению знаний породило новые технологии обработки данных, основанные на различных методах интеллектуального анализа. К таким направлениям научных исследований в области искусственного интеллекта относятся data mining (раскоп данных), knowledge discovery in databases (обнаружение знаний в базах данных), machine learning (машинного обучения). Их возникновение связано с новым витком в развитии средств и методов обработки различной информации, и такими фундаментальными проблемами искусственного интеллекта как распознавание и прогнозирование. Этот виток обязан пришедшему пониманию, что в накопленной информации содержатся скрытые знания, которые можно извлечь и воспользоваться в практических целях. Алгоритмы искусственного интеллекта сильно усугубляют проблему вычислительных ресурсов, так как являются «жадными» алгоритмами и многократно превосходят ресурсоемкость классических алгоритмов поиска, сортировки и т.д.
3. Борьба за эффективное использование вычислительных ресурсов привела к появлению технологий распределенных систем и параллельных вычислений. Использование суперкомпьютеров в рамках подхода распараллеливания вычислительной задачи позволяет получить необходимую производительность, но из-за дороговизны оборудования этот способ может оказаться экономически неэффективным. Получить практически те же вычислительные мощности, что и на суперкомпьютерах, но с гораздо меньшей стоимостью, можно более полно

используя вычислительные ресурсы, имеющегося в распоряжении парка компьютеров, создав grid-систему. Применение grid систем обусловлено относительно низкой стоимостью оборудования, простотой развёртывания и возможностью масштабирования. Grid-система строится по принципу «клиент-сервер» и состоит из одного или нескольких компьютеров-серверов и множества компьютеров-клиентов свободной конфигурации. Компьютеры-клиенты занимаются вычислениями. Функции компьютера-сервера заключаются в выделении каждому из клиентов части вычислений, приеме и агрегировании результатов. Клиенты связываются с сервером с помощью высокоуровневого протокола HTTP.

В рамках проводимых работ по тематике «Разработка облачного ресурса интеллектуального анализа данных» была заложена концепция grid-систем. Поиски путей повышения производительности вычислительной техники [3], особенно при создании библиотеки WEB программ алгоритмов распознавания и прогнозирования (кластеризации и классификации), привели к простому, на первый взгляд, решению — создать распределенную вычислительную систему на базе Internet-технологий. Классические методы интеллектуального анализа данных, хорошо зарекомендовавшие себя в практике их использования и применения, обладают большим потенциалом к распараллеливанию вычислительных процессов и разработки параллельных алгоритмов их реализации. WEB-системы интеллектуального анализа данных разрабатывается как открытая, развивающаяся система, рассчитанная на привлечение по ее разработке студентов. Поэтому в качестве базового языка разработки был выбран наиболее «демократичный» язык PHP.

Облачные вычисления — тот инструментарий, в технологиях которого одним из приоритетов выступает организация распределенных систем и решение задач «распараллеливания» алгоритмов. Задачу организации параллельного вычисления можно решать со следующими типами конфигурации оборудования:

– выполнение на GPU (Open CL);
– выполнение на многопроцессорной системе;
– выполнение на многомашинной системе.

Grid-система предполагает распараллеливание и организацию распределенных вычислений на многомашинной основе с применением Internet-технологий.

Возможность реконфигурировать структуру Internet-соединений переводят задачу повышения эффективности вычислительных ресурсов в русло решения проблем настройки структуры вычислительной сети, подключаемых каналов связи и выделенных серверов в зависимости от исходных алгоритмов и данных и является ничем иным, как реализацией концепции адаптационных вычислительных систем.

Преимущества адаптивных вычислительных систем связаны в основном с тем, что используются следующие приемы.

• Эффективная работа системы определяется, прежде всего, дисциплиной обслуживания, адаптирующей систему к запросам пользователей. При этом задачи ранжируются по ресурсоемкости и под них выделяются соответствующие их рангу вычислительные мощности. Адаптация в данном случае заключается в том, что

дисциплина обслуживания реализует свои функции, не располагая информацией о том, какой длины и сложности задача поступила в вычислительную систему.

- Повышение живучести вычислительной системы путем дублирования ее элементов является типичной адаптацией к ее собственному состоянию, точнее — к состоянию ее элементов. Такая адаптация характерна для систем, которые обычно называются самонастраивающимися. Здесь самонастройка дает возможность вычислительной системе сохранять свою работоспособность независимо от того, что отдельные ее агрегаты в неизвестное заранее время выходят из строя.
- Распараллеливание и конвейер вычислений могут быть использованы не вообще, а лишь применительно к конкретной структуре алгоритма решения задачи. Процесс создания схемы многопроцессорной вычислительной системы — это, прежде всего процесс приспособления, адаптации к структуре выполняемой программы.

Идея распараллеливания применима там, где есть независимые друг от друга участки программы. Даже если программа не имеет таких участков или они очень малы, при массовых расчетах, когда один и тот же алгоритм выполняется многократно, можно сократить общее время вычислений.

Проиллюстрируем эту процедуру на простом примере. Пусть необходимо вычислить сумму для заданных значений аргумента $x_1, x_2, \ldots, x_k$:

$$y = \sum_{i=1}^{n} f_i(x). \tag{1}$$

Хорошо видно, что сначала надо вычислить значения всех функций f_i, а уж потом их сложить. Вычисление функций f_i можно делать параллельно на n серверах, на что потратится время, необходимое для вычисления самой трудоемкой функции f_i, и потом сложить n чисел на центральном сервере. На рис.1 показана структура такой параллельной вычислительной системы из n серверов $S_1, S_2, \ldots, S_n$, на каждый из которых подается программа вычисления «своей» функции f_j (она обозначена $P(f_j)$) и значение аргумента x_i. На выходе j-ого сервера S_j образуется значение $f_j(x_i)$. Все выходы этих серверов суммируются на n+1-ом сервере (центральном) S_{n+1}. Его программа $P(\sum)$ и образует искомую функцию.

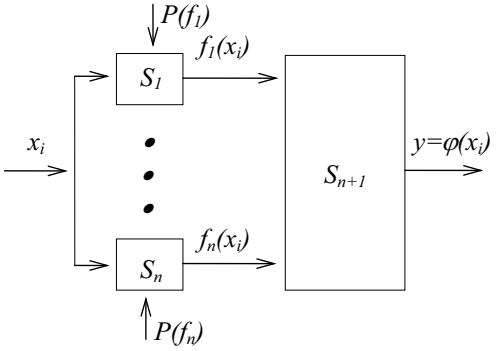

Рис. 1. Реализация параллельного вычисления с помощью нескольких процессоров.

Таким образом, n+1 процессор образовали вычислительную систему, которая вычисляет функцию $\varphi(x_i)$ за время

$$t_{\max} + (n-1)t_{\Sigma}, \tag{2}$$

где $t_{\max} = \max_{i=1}^{n} (t_i)$ — время вычисления самой трудоемкой функции из f_i, а t_{Σ} — время суммирования двух чисел процессором.

При последовательном вычислении этой функции на одном процессоре понадобилось бы время:

$$\sum_{j=1}^{n} t_j + (n-1)t_{\Sigma}, \tag{3}$$

где t_j — время вычисления j-й функции f_j, т.е. значительно больше. Выигрыш от параллельных вычислений, как видно, достаточно велик.

Важнейшей характеристикой системы является время коммутации и передачи данных, которое может оказать значительное влияние на длительность решения задачи. Фактически, интерес представляет время, через которое после начала процесса вычисления пользователь получит его результат. Время получения результата на i-ом компьютере-клиенте (t_i) складывается из времени подключения (*connecting*) (tc_i) и времени на обработку (*processing*) данных (tp_i).

$$t_i = t\text{п}_i + t\text{о}_i. \tag{4}$$

Если время решения задачи на центральном сервере меньше чем время $t_{\max}$ (коммутации плюс время решения на обработку) то распараллеливание не повысит общую производительность системы. Такая ситуация может возникнуть при значительном влиянии характеристик канала связи, в том числе времени коммутации и передачи данных ($tc_i << tp_i$).

Минимизировать tp_i можно за счет установки высокопроизводительного оборудования. Минимизация $t_{\max}$ достигается путём установки дополнительных компьютеров-клиентов, тем самым снижая среднее значение tp_i, а также стремлением к максимально низкому значению tc_i, размещая компьютер-сервер и компьютеры-клиенты в одной локальной сети. Следует помнить, что при tpi близким к tc_i требуется решить задачу выбора оптимального количества компьютеров-клиентов, дабы избежать отрицательного эффекта от применения системы. Оптимизировать параметры tc_i и tp_i стоит исходя из соображений целесообразности и общей эффективности.

При поступлении задачи на решение многопроцессорной вычислительной системой, последняя сначала должна сделать анализ структуры на предмет ее распараллеливания и далее определить конфигурацию вычислительной системы для решения поступившей задачи, т.е. какие ее части на каких процессорах и когда будут решаться. Минимальность времени ее решения будет гарантироваться максимальной загрузкой всех процессов вычислительной системы, о чем и должен заботиться супервизор

grid-системы при составлении плана решения каждой задачи. Для обеспечения такой задачи, важным компонентом grid-системы выступает модуль, позволяющий анализировать ресурсы (компьютеры и серверы), подключенные (доступные) к системе на базе Internet-сети.

Ниже представлено описание реализации на языке PHP алгоритма такого анализа.

1. На вход программы поступают исходные данные — тестируемая конфигурация (адреса подключаемых удаленных серверов), исследуемая формула вычисления (в рамках синтаксиса PHP) и тестовый диапазон чисел.
2. Компьютер-сервер разбивает весь диапазон значений на количество поддиапазонов, соответствующее количеству подключенных к системе компьютеров-клиентов, участвующих в распараллеливании.
3. Каждый компьютер-клиент по протоколу HTTP получает диапазон значений для обработки.
4. Система ожидает завершения подсчета на каждом компьютере-клиенте и агрегирует результаты вычислений.
5. На выходе программы — время, затраченное на обработку задания в вычислительной системе, а также массив данных с табличной структурой с указанием каждого сервера, временем обработки на нем функции, временем коммутации и передачи данных.

Разработка grid-системы с возможностью наращивания вычислительных решений за счет Internet-подключений (серверов), снабженная инструментарием управления, распределения и конфигурирования ресурсов выводит на новый уровень применения таких сложных, ресурсоемких автоматизированных систем как системы интеллектуального анализа данных, экспертные системы, системы прогнозирования. Важнейшей составляющей такого инструментария выступает решение задач планирования и распределения, основанных на принципах распараллеливания и адаптивной настройки вычислительных ресурсов к вычислительной задаче, структуре процесса обработки и структуре данных.

Литература

1. *Бескид П.П., Шишкин А.Д.* Об опыте проведения мониторинга состояния морской поверхности радиолокационными средствами. // Безопасность жизнедеятельности, 2011, № 2, с. 20–24.
2. *Сикулер Д.В., Фомин В.В.* Концепция Internet–системы интеллектуальной обработки данных. Некоторые актуальные проблемы современной математики и математического образования. Герценовские чтения — 2011. Материалы научной конференции, 11–16 апреля 2011 г. — СПб., 2011, с. 206–209.
3. *Флегонтов А.В., Фомин В.В.* Система интеллектуальной обработки данных. // Известия Российского государственного педагогического университета им. А.И. Герцена, 2013, № 154, с. 41–48.