Г. И. Борзунов, А. Е. Войнов, Т. В. Петрова

АНАЛИЗ МЕТОДОВ ПОВЫШЕНИЯ ЭФФЕКТИВНОСТИ РАСПРЕДЕЛЕННЫХ ВЫЧИСЛЕНИЙ ПРИ РЕШЕНИИ ЗАДАЧ БЕЗОПАСНОСТИ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

В настоящее время все большее значение как в реализации криптосистем [1], так и в исследованиях вычислительной стойкости криптоалгоритмов [2] приобретают распределенные вычисления. Одним из стратегических направлений развития современных компьютерных технологий является широкое использование кластеров (clusters). Под кластером обычно понимается (см., например, [3]) множество отдельных компьютеров, объединенных в сеть при помощи специальных аппаратно-программных средств. Важным преимуществом параллельных программ, разработанных с использованием кластера, является их мобильность. Так, при решении задач большой временной сложности программы, разработанные для кластера с использованием MPI-технологии, могут быть перенесены практически без доработки на более производительный суперкомпьютер, например на Многопроцессорную вычислительную систему Межведомственного суперкомпьютерного центра РАН [3]. Вместе с этим следует отметить, что организация взаимодействия вычислительных узлов кластера при помощи передачи сообщений может приводить к значительным временным задержкам, что накладывает дополнительные ограничения на тип разрабатываемых параллельных алгоритмов и программ. Таким образом, для эффективной реализации распределенных вычислений требуется обеспечить равномерную вычислительную нагрузку для всех процессоров кластера и минимальные информационные потоки передачи данных между этими процессорами [4, 5]. Естественной моделью решения этой задачи является граф, вершины которого соответствуют подзадачам и взвешены временной сложностью этих подзадач, а ребра представляют собой обмен сообщениями между подзадачами и взвешены соответствующими коммуникационными затратами. В терминах этой модели решение указанной выше задачи сводится к разбиению вершин графа на непересекающиеся подмножества с максимально близкими значениями суммарных весов вершин подмножеств и при минимальном значении суммы весов ребер, соединяющих вершины из разных подмножеств. В качестве пространства решений задачи рассматривается множество всех допустимых разбиений графа на непересекающиеся подграфы. Допустимыми разбиениями графа считаются такие разбиения, для которых выполняются некоторые ограничения, определяемые предметными областями решаемых задач. Из множества допустимых разбиений выделяется оптимальное разбиение. Критериями оптимальности могут быть: 1) равенство сумм весов вершин подграфов, 2) минимальность суммы весов ребер, соединяющих вершины, принадлежащие разным подграфам; 3) число подграфов. Первый критерий определяется временной сложностью

подзадач, решение которых осуществляется на разных процессорах. Смысл второго критерия состоит в том, что он обуславливает минимум коммуникационных затрат. Наконец, число подграфов определяет число используемых процессоров. Постановки задач оптимального разбиения графов могут варьироваться в зависимости от свойств графов, которые определяются предметной областью задачи. Известно, что проблема оптимального разделения взвешенного графа является NP-сложной и точные методы ее решения существуют для очень узкого класса задач. Поэтому актуальным является алгоритмический анализ оптимального разделения графов и определение подходов, позволяющих повысить эффективность алгоритмов решения этой задачи. Ниже для теоретического исследования методов разделения графов используются методы теории анализа алгоритмов. На практике обычно оптимальное разделение графа выполняется для некоторого частного случая, например: 1) веса вершин и ребер графа равны 1, и в качестве приоритетного критерия используется либо равенство суммарных весов вершин, принадлежащих одному и тому же подмножеству, либо условие минимальной суммы разрезанных ребер; 2) веса вершин различны, веса ребер равны 0, и в качестве критерия оптимальности используется только условие равенства суммарных весов вершин, принадлежащих одному и тому же подмножеству. В последнем частном случае задача оптимального разделения графа сводится к задаче оптимального разбиения множества [6], но все еще остается NP-полной. В данной работе рассматриваются наиболее часто используемые подходы к решению практически значимых частных случаев задачи оптимального разделения графа. Так, для разделения графов, моделирующих решение задач, в которых области расчетов аппроксимируются двумерными или трехмерными вычислительными сетками, используются геометрические методы [3]: покоординатное разбиение, рекурсивный инерционный метод деления пополам, деление сети с использованием кривых Пеано. Эти методы основываются на координатной информации об узлах графа.

Метод покоординатного разбиения предполагает рекурсивное деление графа пополам по наиболее длинной стороне. Такой способ разбиения дает существенно меньшее количество информационных связей между разделенными частями по сравнению со случаем, когда граф делится по меньшей стороне. Общая схема выполнения метода состоит из следующих действий: 1) вычисляются центры масс элементов сети; 2) полученные точки проектируются на ось, соответствующую наибольшей стороне разделяемой сети, и получаем упорядоченный список всех элементов сети; 3) делим полученный список пополам и в результате получаем требуемое разделение (бисекцию).

Метод координатного разбиения работает очень быстро и требует небольшого количества оперативной памяти. Однако получаемое разбиение уступает по качеству более сложным и вычислительно трудоемким методам. Кроме того, в случае сложной структуры графа алгоритм может получать разбиение с несвязанными подграфами. Временная сложность алгоритма равна O(N). Тем не менее нередко применение этого алгоритма не дает хороших разбиений. Например, приведенная выше схема может производить разбиение графа только по линии, перпендикулярной одной из координатных осей. Во многих случаях такое ограничение оказывается критичным для построения качественного разбиения. Достаточно повернуть граф под острым углом к координатным осям, чтобы убедиться в этом. Для минимизации границы между подграфами желательна возможность проведения линии разделения с любым требуемым углом поворота.

Возможный способ определения угла поворота, используемый в рекурсивном инерционном методе деления пополам, состоит в использовании главной инерционной оси [7], считая элементы графа точечными массами. Линия бисекции, ортогональная полученной оси, как правило, дает границу наименьшей длины. Временная сложность алгоритма равна O(N). Алгоритм дает несколько лучшие результаты, чем покоординатное разбиение, но все еще недостаточно стабилен на неоднородных или вырожденных графах.

Одним из недостатков рассмотренных выше графических методов является то, что при каждой бисекции эти методы учитывают только одну размерность. Схемы, учитывающие больше размерностей, могут обеспечить лучшее разбиение. Один из таких методов упорядочивает элементы в соответствии с позициями центров их масс вдоль кривых Пеано [8]. Кривые Пеано — это кривые, полностью заполняющие области больших размерностей (например, квадрат или куб). Применение таких кривых обеспечивает близость точек фигуры, которые соответствуют точкам, близким на кривой. После получения списка элементов графа, упорядоченного в зависимости от расположения на кривой, достаточно разделить список на необходимое число частей в соответствии с установленным порядком. Результаты развития описанных выше геометрических методов разделения графов приводятся в работах [9–12]. Таким образом, геометрические методы выполняют разбиение графов, моделирующих сеть, основываясь исключительно на координатной информации об узлах графа. Так как геометрические методы не принимают во внимание информацию о связности элементов сети, то они не могут явно привести к минимизации суммарного веса граничных ребер. Для минимизации межпроцессорных коммуникаций геометрические методы оптимизируют некоторые вспомогательные показатели (например, длину границы между разделенными участками графа). Геометрические методы работают очень быстро (временная сложность почти во всех случаях равна O(N)). Можно рекомендовать использовать геометрические методы для оптимизации распределенных вычислений на сетках различной размерности. Однако ввиду указанных выше особенностей применение этих методов в других областях, например в исследованиях вычислительной стойкости криптоалгоритмов, представляется малоэффективным.

Метод спектральной бисекции (спектрального деления пополам) обеспечивает без учета веса вершин достаточно высокое качество разбиения графа [13]. Применяя этот метод рекурсивно, можно разбить граф на произвольное число частей. Пусть задан неориентированный взвешенный граф G(V,E) с числом вершин, равным n: $V=\{v_1,\ldots,v_n\}$, и числом ребер, равным m: $E=\{e_1,\ldots,e_m\}$. В соответствие каждой вершине v_i ставится координата характеристического вектора x[i], значение которой равно +1 или -1. Значения координат характеристического вектора выбираются таким образом, что их общая сумма равна 0. Значение координат характеристического вектора x[i], равное +1, определяет, что вершина v_i первоначально принадлежит первому подмножеству разбиения вершин графа V_1, а значение x[j] = -1 относит вершину v_j ко второму подмножеству V_2 того же разбиения вершин графа. Таким образом, вектор x[] определяет разбиение множества вершин графа на два непересекающихся подмножества. Второе свойство вектора x[] (равенство 0 суммы координат) обеспечивает равную мощность подмножеств разбиения. Определим теперь R как функцию от вектора x[] :

$$R=\sum F2\ C(V_1,V_2) = f(x[]) = (1/4)\sum_E (x[i]-x[j])^2 . \qquad (1)$$

В выражении 1 предполагается суммирование всех элементов $(x[i]-x[j])^2$, соответствующих ребрам графа G, и только этих элементов. Справедливость этого равенства следует из того, что если x[i] и x[j] принадлежат одному и тому же подмножеству вершин, то $(x[i]-x[j])^2 = 0$, иначе $(x[i]-x[j])^2 = 4$.

Теперь имеется функция для минимизации, преобразуем ее к матричной форме, используя матрицу Лапласа. Так как разбиение графа — NP-трудная задача, целесообразно ослабить ограничения дискретности на x[] и сформулировать новую непрерывную задачу:

$$\text{Минимизировать}: \quad f(x) = \frac{1}{4} x^t L x$$

$$\text{Ограничения}: \quad x^t 1 = 0, \quad x^t x = n,$$

где 1 — это n-мерный вектор $(1,1,1,1,..)^t$, n — число вершин графа. Матрица Лапласа L имеет вид:

$$L_{ij} = \begin{cases} -1, & \text{если вершины i и j соединеы ребром} \\ di, & \text{если } i = j \\ 0, & \text{иначе} \end{cases},$$

где di — это число ребер, инцидентных i-й вершине.

Эта непрерывная задача — только приближение к дискретной, и значения, определяющие ее решение, должны быть отображены обратно в множество {+1, -1}. Идеально, когда решение оказывается близким к значениям +1, -1. Справедлива следующая теорема: если U_1, U_2 … — нормализованные собственные векторы матрицы L с соответствующими собственными значениями $\lambda_1 <= \lambda_2 <= \lambda_3$… то матрица L имеет следующие свойства: 1) L — симметрична; 2) U_i попарно ортогональны; 3) $U_1 = n^{-0,5}1$, $\lambda_1 = 0$; 4) если граф связный, то только λ_1 принимает нулевое значение. Выразим вектор x[] в терминах собственных векторов матрицы L: $x[i] = \Sigma\alpha_i U_i$, где α_i — вещественные константы, такие, что $\Sigma(\alpha_i)^2 = n$. Свойство 2) гарантирует, что это всегда возможно. Подставляя в выражение f(x[]) новое значение x[] и учитывая, что $\lambda_1 = 0$, получаем функцию для минимизации, зависящую от собственных значений матрицы Лапласа, начиная λ_2 и заканчивая λn: $f(x[]) = 0{,}25(\alpha_2^2 \lambda_2 + \alpha_3^2 \lambda_3 + \ldots + \alpha_n^2 \lambda n)$. Очевидно, что выполняется неравенство;

$(\alpha_2^2 + \alpha_3^2 + \ldots + \alpha_n^2)\ \lambda_2 <= (\alpha_2^2 \lambda_2 + \alpha_3^2 \lambda_3 + \ldots + \alpha_n^2 \lambda n)$, из которого с учетом упорядоченности собственных значений матрицы L следует справедливость неравенства $f(x[]) >= n\ \lambda_2/4$. Таким образом, можно минимизировать функцию $f(x[]) = n\ \lambda_2/4$, выбирая ее аргумент равным: $x[] = n^{0,5} U_2$. Полученный в результате указанных вычислений вектор x[] представляет собой решение непрерывной задачи. Теперь для получения характеристического вектора, определяющего разделение множества вершин графа на два подмножества, необходимо отобразить координаты полученного вектора x[] в множество ={-1, +1}. Для этого находится медиана значений x[], и затем все координаты вектора x[], имеющие индекс, меньший или равный индексу медианы, принимают значения, равные -1, а координаты с индексом, превосходящим индекс медианы, принимают значения, равные +1. Это решение является самой близкой точкой в многомерном дискретном пространстве к непрерывному оптимуму. В работах [14—16] описываются алгоритмы, представляющие собой обобщение алгоритма спектральной бисекции: спектральное деление исходного графа на четыре подграфа (quadrisection) и спектральное деление исходного графа на восемь подграфов (octasection). Эти алгоритмы соответственно делят исходный граф на четыре подграфа, используя два собственных вектора матрицы L, и на восемь подграфов, используя три собственных вектора матрицы L. Результаты, полученные с использованием этих алгоритмов, превосходят результаты, полученные с помощью базового спектрального деления пополам. Таким образом, спектральные методы целесообразно применять для минимизации коммуникационных затрат при выполнении распределенных вычислений с использованием кластеров. Существенным недостатком этого метода является сложность определения (например, с помощью метода Ланцоша) компонент вектора Фидлера вырожденной спектральной матрицы графа, что ограничивает число вершин разбиваемого графа. Указанные трудности могут быть преодолены с помощью многоуровневых алгоритмов [15, 16].

Из приведенного выше анализа следует, что геометрические методы обладают рядом преимуществ при расчетах баланса вычислительной нагрузки процессоров без учета коммуникационных затрат, а спектральные методы минимизируют коммуникационные затраты при существенных ограничениях на структуру графа. В общем случае минимизация коммуникационных затрат и равномерное распределение вычислительной нагрузки по процессорам являются противоречивыми требованиями. Поиск компромиссного решения требует применения комбинаторных методов, которые будут рассмотрены в дальнейших статьях.

СПИСОК ЛИТЕРАТУРЫ:

1. *Pearson D.* A Parallel Implementation of RSA. Computer Science Department. Cornell University, Ithaca, NY 14853. July 22, 1996.
2. *Бабенко Л. К., Курилкина А. М.* Распараллеливание криптоаналитического метода «разделяй и побеждай» для каскадных шифров // Материалы XII Всероссийской научно-практической конференции «Проблемы информационной безопасности в системе высшей школы». М.: МИФИ, 2005. С. 14–15.
3. *Гергель В. П.* Теория и практика параллельных вычислений. М.: БИНОМ. Лаборатория знаний, 2007. – 423 с.
4. *Хьюоз К., Хьюоз Т.* Параллельное и распределенное программирование с использованием C++. М.: Издательский дом «Вильямс». 2004. – 672 с.
5. *Нарайкин А. А., Лопатин И. В.* Проблемы эффективного использования узлов на основе архитектуры INTEL в гетерогенных кластерах // Материалы Международного научно-практического семинара «Высокопроизводительные параллельные вычисления на кластерных системах» / Под ред. проф. Р. Г. Стронгина. Нижний Новгород: Изд-во Нижегородского университета, 2002. С. 124.
6. *Романовский И. В.* Алгоритмы решения экстремальных задач. М.: Главная редакция физико-математической литературы изд-ва «Наука», 1977. С. 247–251.
7. *Pothen A.* Graph partitioning algorithms with applications to scientific computing. Kluwer Academic Press, 1996.
8. *Oui C., Ranka S., Fox G.* Fast and parallel mapping algorithms for irregular and adaptive problems // Journal of Supercomputing. 1996. 10. P. 119–140.
9. *Patra A., Kim D.* Efficient mesh partitioning for adaptive hp finite element methods // International Conference on Domain Decomposition Methods. Greenwich, U.K., August, 1998.
10. *Pilkington J., Baden S.* Partitioning with space filling curves. Technical Report CS94-349, Dept. of Computer Science and Engineering, Univ. of California. 1994.
11. *Miller G. L., Teng S.-H., Thurston W., Vavasis S. A.* Automatic mesh partitioning // George A., Gilbert J. R., Liu J. W. H., eds., Graph Theory and Sparse Matrix Computation, Springer-Verlag, 1993.
12. *Nicol D. M.* Rectilinear partitioning of irregular data parallel computations. Tech. Rep. 91--55, ICASE, NASA Langley Res. Ctr., Jul. 1991.
13. *Hendricson B., Leland R.* Multidemensional spectral load balancing. Rep. SAND93-0074, Sandia National Laboratories, Albuquerque, NM, January 1993.
14. *Bradford L.* Chamberlain. Graph Partitioning Algorithms for Distributing Workloads of Parallel Computations. 1998.
15. *Бувайло Д. П., Толок В. А.* Быстрый высокопроизводительный алгоритм для разделения нерегулярных графов // Вісник Запорізького державного університету. 2002. № 2. С. 1–10.
16. *Якобовский М. В.* Обработка сеточных данных на распределенных вычислительных системах // Вопросы атомной науки и техники. Сер. Математическое моделирование физических процессов. 2004. Вып. 2. С. 40–53.