Исследование эффективности параллельных алгоритмов раскраски графа для распределенных систем

А.Е. Шухман, А.В. Сериков

Пусть $G=(V,E)$ - неориентированный граф. Произвольная функция $f:V\to\{1,2,...,k\}$, где k принадлежит множеству натуральных чисел, называется вершинной k - раскраской графа G. Раскраска называется правильной, если $f(u)\neq f(v)$, для любых смежных вершин u и v. Граф, для которого существует правильная k - раскраска, называется k - раскрашиваемым. Минимальное число k, при котором граф G является k - раскрашиваемым, называется хроматическим числом графа и обозначается $\chi(G)$ [1]. Поиск правильной k - раскраски графа G, такой что $k=\chi(G)$, является NP-трудной задачей. Жадный приближенный алгоритм окраски весьма эффективный, но строго последовательный, следовательно, на его основе трудно составить параллельный алгоритм.

В данной работе рассматривается алгоритм, предложенный в статье [2]. В наши задачи входило исследование эффективности реализации алгоритма на основе технологии MPI. Алгоритм организован в виде последовательности супершагов. Каждый супершаг включает две фазы: фаза вычисления и фаза коммуникации. Параметр S – размер супершага. В фазе вычисления каждый процессор последовательно красит S вершин, используя информацию о цветах соседних вершин, которая имеется на момент начала супершага. В фазе коммуникации процессоры обмениваются новой информацией о цветах вершин. Две смежные вершины могут быть одновременно окрашены на разных процессорах и получить один и тот же цвет, следовательно, необходимо проводить проверку для выявления конфликтов. В случае обнаружения конфликта одна из вершин должна быть перекрашена на следующем супершаге с учетом новой информации о цветах. Какая из двух вершин будет перекрашена, выбирается случайно, чтобы более равномерно распределять нагрузку на процессоры.

От параметра S зависит как число супершагов, необходимых для решения задачи, так и число конфликтов, возникающих между процессорами. На большинстве тестовых графов лучшие результаты получены при S=25.

Трудно оценить ожидаемое ускорение, так как заранее неизвестно число конфликтов между процессорами, которые возникнут при решении задачи на конкретном графе. При небольшом числе связей между вершинами временные затраты на коммуникацию оказываются выше затрат на поиск цвета для вершины, следовательно доля последовательных действий f окажется больше 0,5. При большом числе связей затраты на поиск цвета для вершины будут выше затрат на коммуникацию, следовательно f окажется меньше 0,5. В случае, когда число ребер оказывается близко к максимальному для данного графа, резко возрастает число конфликтов, число выполняемых итераций увеличивается, ускорение уменьшается. Для теоретической оценки будем считать, что среднее значение f = 0,5. По закону Амдала ожидаем получить ускорение 1,33 для двух процессоров и 1,77 для восьми. На практике ускорение может оказаться как выше, так и ниже в зависимости от конкретного примера. Измерения для двух процессоров показали зависимость ускорения от величины $2|E|/|V|^2$, которая лежит в интервале от 0 до 1 для неориентированного графа. Среднее значение ускорения при $2|E|/|V|^2 < 0{,}025$ составило 1,1, при $0.025 < 2|E|/|V|^2 < 0{,}95$ среднее ускорение 1,42 и при $2|E|/|V|^2 > 0{,}95$ среднее ускорение 1,26. Полученная зависимость может быть использована для оптимизации количества коммуникаций с целью повышения ускорения. В дальнейшем планируется исследовать масштабируемость алгоритма при увеличении количества процессоров и рассмотреть возможность дальнейшей оптимизации реализации алгоритма.

Литература

1. Оре О., Теория графов. - 2-е изд. - М.: Наука, 1980. – 336 с.
2. Erik G. Boman, Doruk Bozdag, Umit Catalyurek, Assefaw H. Gebremedhin, Fredrik Manne, – A Scalable Parallel Graph Coloring Algorithm for Distributed Memory Computers, 2005.