ПРОГРАММНЫЙ КОМПЛЕКС ДЛЯ ГЕНЕРАЦИИ БЕРНСАЙДОВЫХ ГРУПП

С.С. Карчевский

В 1902 году У. Бернсайд сформулировал следующую проблему: "Будет ли всякая группа с конечным числом m образующих и тождественным соотношением (x^n)=1 конечной?". Такие группы обозначаются как B(m,n). В общем случае ответ на поставленный вопрос отрицательный, однако до сих пор остается открытым вопрос о конечности некоторых бернсайдовых групп. Одна из таких - группа с двумя образующими и показателем n=5.

В 1955 году А. И. Кострикин показал, что существует максимальная конечная группа с двумя образующими и n=5(обозначается как B0(2,5)). Позднее было доказано, что ее порядок равен 5^34. Так же известно, что если B(2,5) конечна, то B0(2,5) изоморфна B(2,5). Таким образом, для решения вопроса о конечности B(2,5) достаточно доказать, что она изоморфна B0(2,5). Образующие элементы группы B0(2,5) могут быть представлены в виде матриц размером 66 на 66 с целочисленными элементами из множества {0,1,2,3,4}. При таком представлении групповой операцией является умножение матриц, после которого все элементы результата берутся по модулю 5, а единицей группы является единичная матрица. Очевидно, что группа, порожденная такими матрицами, будет конечна. Для группы B(2,5) на данный момент не существует подобного представления, как и не решен вопрос о ее конечности. Однако, используя концепции минимальных слов и независимых слов, описанные А. А. Кузнецовым в [1], группу B(2,5) можно представить в виде слов, составленных из алфавита из двух элементов. В качестве таких элементов удобно выбрать 0 и 1, групповой операцией будет конкатенация слов, а единицей будет пустое слово. При этом на множестве слов вводится отношение порядка: слово x меньше слова y, если слово x лексикографически младше, при условии, что 0<1. Можно так же определить отображение из множества слов на множество матриц, если поставить одну порождающую матрицу в соответствие младшему элементу алфавита, а другую - старшему, а затем перемножить матрицы в том порядке, в каком следуют элементы алфавита в слове при чтении слева направо. Ясно, что некоторым разным словам будут соответствовать одни и те же матрицы в силу конечности множества матриц с элементами по модулю 5. Пара таких слов называется соотношением. Множество всех слов одной длины называется объектом, при этом длина слова называется номером объекта. Для получения слова из объекта с номером K+1 надо к слову из объекта K приписать слева один из образующих элементов. Чтобы получить весь объект, надо проделать эту операцию для всех слов из K и всех образующих. Далее, надо исключить из полученного объекта слова, являющиеся младшими в соотношениях, имеющихся в этом и всех предыдущих объектах.

Существует критерий[2], позволяющий разбить объект на классы таким образом, что любые два слова, состоящие в соотношении, будут содержаться в одном классе. Более того, при описанном выше расширении объекта(получении объекта K+1 из K) все слова из одного класса будут переходить в один и тот же класс, т.е. достаточно вычислить номер класса только для одного расширенного слова. Таких классов будет всего (5^7)=78125. Это позволяет сильно сузить области поиска соотношений и сократить время. Т.к. соотношения ищутся путем сравнения матриц, для которых не определен порядок, то для поиска нельзя использовать алгоритмы, требующие предварительной сортировки, т.е. остается только перебрать. Таким образом, каждый класс можно обрабатывать независимо от остальных, т.е. параллельно.

Для реализации предложенных идей написана программа с использованием MPI и работающая по клиент-серверному принципу. На каждом большом этапе (расширение, поиск соотношений, фильтрация) на всех процессах, кроме нулевого (в смысле рангов MPI), запускается рабочая функция, а на нулевом - серверная, для синхронизации. В общем случае рабочие процессы запрашивают у нулевого номер необработанного еще класса, нулевой сообщает номер рабочему и инкрементирует счетчик обработанных классов. Каждый класс хранится в отдельном файле, а на этапе расширения каждая из двух получающихся частей нового класса записывается в отдельный файл. На этапе отображения слов на матрицы используются таблицы умножения, т.е. предварительно вычисленные матрицы некоторых слов длинны, меньшей чем K. Сами таблицы умножения хранятся в виде деревьев, поиск по которым проходит за q шагов, где q - длина слова, матрица которого ищется.

На данный момент все процессы равноправны, что неэффективно на кластерах с многопроцессорными узлами. Поэтому до осени планируется сделать многопоточность в рамках одного узла вместо многопроцессности, что позволит учитывать особенности вычислительной установки, на которой запускается программа. Если это будет реализовано, то на каждом многопроцессорном узле планируется запустить по одному сетевому потоку, который будет слушать некоторый порт, а аналогичный поток нулевого процесса будет подключаться ко всем узлам по очереди через определенные интервалы времени. Главным образом такая организация будет служить для своевременного предупреждения о зависании узла или его выходе из строя, т.к. опыт показал, что MPI выдает предупреждения о прервавшихся процессах только в редких случаях, когда обрыв происходил непосредственно во время передачи. Дополнительно по этому сетевому подключению можно будет передать служебную информацию о том, что происходит внутри процесса на каждом конкретном узле. Это позволит точно прогнозировать время завершения программы. Полученный прогноз планируется выдавать по запросу программы-клиента, которая будет подключатся по сети ко второму сетевому потоку главного процесса. Было решено отказаться от вывода прогноза и служебной информации в файл, т.к. программа и так интенсивно использует дисковую систему независимыми обращениями каждого процесса, а информация нужна и актуальна только на момент вывода, т.е. хранить ее не нужно. Более того, на вычислительной установке, как правило, работают несколько других пользователей, процессы которых так же интенсивно обращаются к системе хранения.

В настоящее время программа работает на 560 ядрах на кластере Сибирского Федерального Университета[3] и обрабатывает объект номер 36.

ЛИТЕРАТУРА:

1. Кузнецов А. А. - Комплекс алгоритмов компьютерного моделирования дискретных алгебраических систем, Красноярск 2009.
2. Havas~G., Wall~G., Wamsley~J. The two generator restricted Burnside group of exponent five // Bull. Austral. Math. Soc. 1974. \No~10. P.~459--470.
3. http://cluster.sfu-kras.ru/
4. Кострикин А. И. - Вокруг Бернсайда, М.: Наука, 1986