МГТУ им. Н. Э. Баумана

ИУ 7 – 32

Отчет о лабораторной работе №6

«Обработка разреженных матриц»

Вариант №8

Исаев Д.С.

# Цель работы

Реализация алгоритмов обработки разреженных матриц, сравнение этих алгоритмов со стандартными алгоритмами обработки матриц.

# Условие задачи

Разреженная (содержащая много нулей) матрица хранится в форме 3-х объектов:

- вектор ***A*** содержит значения ненулевых элементов;

- вектор ***JA*** содержит номера столбцов для элементов вектора ***A***;

- связный список ***IA***, в элементе Nk которого находится номер компонент

в ***A*** и ***JA***, с которых начинается описание строки Nk матрицы ***A***.

1. Смоделировать операцию умножения хранящихся в этой форме матрицы и вектора-столбца с получением результата в той же форме.

2. Произвести операцию умножения, применяя стандартный алгоритм работы с матрицами.

3. Сравнить время выполнения операций и объем памяти при использовании этих 2-х алгоритмов при различном проценте заполнения матриц.

**Описание**

**Разреженные матрицы**

Это матрицы, содержащие большой процент нулевых элементов.

Такие матрицы часто встречаются в математике, например при представлении неориентированных графов.

**Реализация**

Разреженная (содержащая много нулей) матрица хранится в форме 3-х объектов:

- вектор ***A*** содержит значения ненулевых элементов;

- вектор ***JA*** содержит номера столбцов для элементов вектора ***A***;

- связный список ***IA***, в элементе Nk которого находится номер компонент

в ***A*** и ***JA***, с которых начинается описание строки Nk матрицы ***A***.

Разреженная матрица занимает О(N) памяти, где N – количество ненулевых элементов.

Обычная матрица занимает O(m\*n) памяти, где m, n – размеры матрицы.

Описание структуры разреженной матрицы в программе:

   class SparseMatrix

    {

        Double[] Elements;

        int[] ColumnIndexes;

        List<int> RowsBegin;

…

}

**Результаты**

В результате работы программы были получены следующие данные:

**100% ненулевых элементов:**

* Время умножения для обычных матриц (100% ненулевых элементов): 1,327;
* Память: 1000008 байт на матрицу.
* Время умножения для разреженных матриц (100% ненулевых элементов): 1,201;
* Память: 2001000 байт на матрицу.
* Умножение обычных матриц медленнее на : 10%
* Обычная матрица занимает памяти больше на : -50%

**70% ненулевых элементов:**

* Время умножения для обычных матриц (70% ненулевых элементов): 1,327;
* Память: 1000008 байт на матрицу.
* Время умножения для разреженных матриц (70% ненулевых элементов): 0,954;
* Память: 1399378 байт на матрицу.
* Умножение обычных матриц медленнее на : 39%
* Обычная матрица занимает памяти больше на : -28%

**40% ненулевых элементов:**

* Время умножения для обычных матриц (40% ненулевых элементов): 1,322;
* Память: 1000008 байт на матрицу.
* Время умножения для разреженных матриц (40% ненулевых элементов): 0,708;
* Память: 801130 байт на матрицу.
* Умножение обычных матриц медленнее на : 86%
* Обычная матрица занимает памяти больше на : 24%

**10% ненулевых элементов:**

* Время умножения для обычных матриц (10% ненулевых элементов): 1,327;
* Память: 1000008 байт на матрицу.
* Время умножения для разреженных матриц (10% ненулевых элементов): 0,455;
* Память: 200538 байт на матрицу.
* Умножение обычных матриц медленнее на : 191%
* Обычная матрица занимает памяти больше на : 398%

Из приведенных результатов видно, что чем больше нулевых элементов в матрице, тем более выгодно представлять ее в разреженной форме. При 40% ненулевых элементов выигрыш в скорости составляет 86%, при 10% уже 191%. К тому же обычная матрица занимает значительно больше памяти при наполненности нулевыми элементами более половины матрицы.

**Вывод:**

При наполненности матрицы нулевыми элементами(50% и выше) значительно выгоднее использовать разреженные матрицы и по скорости и по размеру используемой памяти.