

Решение СЛАУ методом LDU-разложения

Николай Жидков

15 марта 2018 г.

1 Структура программы

Программа разделена на функции, записанные в файле `solve.py`. Основных функций 3, остальные должны быть понятны из названий

- *read(filename)*, функция чтения:
 - Принимает название файла для чтения данных
 - Возвращает размерность n , матрицу A и столбец b .
- *solve(n, A, b, full_mode)*, решает СЛАУ:
 - Принимает размерность n , матрицу A , столбец b , включен ли режим подробного вывода
 - Возвращает решение системы $Ax = b$.
- *noise(n, A, b, x, noise_experiments, full_mode)*, проводит численный эксперимент:
 - Принимает размерность n , матрицу A , столбец b , столбец решений x , количество повторений эксперимента, включен ли режим подробного вывода
 - Выписывает все результаты в *stdout*.

2 Структура файлов исходных данных

Исходные данные вводятся из файла, состоящего из n строк, числа на строках разделяются пробелами. Строка i имеет следующую структуру: $A_{i,0}, A_{i,1}, \dots, A_{i,n}, b_i$. Пример содержимого файла для системы третьего порядка:

Пример содержимого файла для системы третьего порядка:

```
1 2 3 4
4 5 6 5
7 8 9 10
```

3 Примеры вызова из командной строки

- Решение СЛАУ, выводится только ответ (файл с входными данными обязательно указывать первым параметром!)

```
python3 solve.py input.txt
```

- Решение СЛАУ, выводится вся дебаг информация

```
python3 solve.py input.txt -full
```

- Численный эксперимент (в примере проводятся 10)

```
python3 solve.py input.txt -noise10
```

- Численный эксперимент со всей дебаг информацией

```
python3 solve.py input.txt -noise10 -full
```

4 Тесты

- Файл вида `pxn.txt` — тест нормального выполнения алгоритма, все условия применимости выполнены (выданы преподавателем).
- Файл `det0.txt` — тест аварийного завершения (определитель системы 0)
- Файл `swap.txt` — тест проверки работы алгоритма при необходимости перестановок

5 Численный эксперимент

5.1 Результаты на СЛАУ 6x6

Решение:

$X = [-0.3, -0.6, 0.8, -1, 0.1, 0.1]$

Число проведенных расчетов: 100

	Возмущение матрицы (%)	Возмущение вектора (%)	Чувствительность решения
максимальное	1.94037	36206.79105	20474.26305
среднее	1.22290	1317.72694	1063.74436
минимальное	0.21984	95.66139	68.31059

5.2 Результаты на СЛАУ 7x7

Решение:

$X = [0, 1.1, 0.8, -1, 0, -0.1, 1.3]$

Число проведенных расчетов: 100

	Возмущение матрицы (%)	Возмущение вектора (%)	Чувствительность решения
максимальное	1.84549	19.75074	14.48756
среднее	1.36498	6.32435	4.70873
минимальное	0.96996	1.24472	0.95089

5.3 Выводы

- Решение СЛАУ 1 мало сильно чувствительна к возмущениям.

Матрица имеет слишком большое число обусловленности, надо с помощью методов с пары привести систему к эквивалентной с меньшим числом обусловленности

- Решение СЛАУ 2 слабо чувствительно к возмущениям.

Рекомендации

6 Замечание

Данный алгоритм LDR-разложения был адаптирован с алгоритма LU-разложения, рассказанного на паре, вместе с перестановкой строк при необходимости. Получилось, что в худшем случае, если на каждом шаге $d_{i,i} = 0$ и мы будем пытаться ставить каждую строку на место i -ой, то время работы будет $O(n^4)$. В реальности такой контрпример подобрать сложно, поэтому почти всегда алгоритм работает нужные $O(n^3)$. Если все-таки нужен алгоритм за $O(n^4)$ в любом случае, то напишите, я подумаю, можно ли все-таки оптимизировать.