Мат. методы теор. систем, лаб. раб. 5, Лихачев В.А., ВТм-112, вариант 6

Задание: решить систему уравнений с использованием якобиана и без. Сравнить результаты.

Генерация исходных данных:

```
function [A out] = generate_data(N)
x = sym(zeros(1, N));
for i=1:N
    x(i) = sym(sprintf('x%d', i));
end

A = sym(zeros(N, N + 1));
for row=1:N
    for col=1:N
        if row == col
            A(row, col) = randn()* sym(x(col)^3);
        elseif col == N
            A(row, col) = randn() * sym(sprintf('cos(x%d)', N));
        elseif col == row - 1
            A(row, col) = randn() * sym(x(col)^2);
        else
            A(row, col) = randn() * sym(x(col));
        end
    end
end
for row=1:N
    A(row, N + 1) = rand();
end
A(1, 2) = rand() * x(2)^2;
out = sym(zeros(N, 1));
jac = sym(zeros(N, N));
for row=1:N
    for col=1:N+1
        out(row) = out(row) + A(row, col);
    end
end
```

```
for row=1:N
    for col=1:N
        jac(row,col)=diff(out(row),x(col));
    end
end
setappdata(0, 'x', x);
setappdata(0, 'out', out);
setappdata(0,'jac',jac);
end
```

Решение системы без использования якобиана:

```
function F = big_matr(in_x)
out = getappdata(0, 'out');
x = getappdata(0, 'x');
F = [subs(out, x, in_x)];
end
```

Решение системы с использованием якобиана:

```
function [F, J] = big_matr_jac(in_x)
out = getappdata(0, 'out');
x = getappdata(0, 'x');
jac=getappdata(0, 'jac');
F = [subs(out, x, in_x)];
J =[subs(jac,x,in_x)];
end
```

Решение системы размерностью 10:

```
N = 10;
generate_data(N);
x0 = randn(N, 1);
options = optimset('Display','iter', 'MaxFunEvals', 3000, 'MaxIter', 1000);
tic; [x1, fval1, exitflag1] = fsolve(@big_matr, x0, options); toc;
```

Iteration	Func-count	f(x)	Norm of step	First-order optimality	Trust-region radius
0	11	143.886		52.9	1
1	22	54.0687	1	15.8	1
2	33	24.3459	2.5	30	2.5
3	44	0.546281	0.939948	1.16	2.5
4	55	0.0740997	0.774046	0.446	2.5
5	66	0.0020025	0.282759	0.0328	2.5
6	77	0.000113752	0.159895	0.00688	2.5

```
     7        88    3.57927e-006      0.0713175         0.0012            2.5
     8        99    1.56366e-008      0.0186663      8.92e-005            2.5
     9       110    4.66317e-013     0.00138452      5.01e-007            2.5
Elapsed time is 29.959938 seconds.
x1 =
    0.0911
    0.1045
    0.4188
   -0.5588
    0.2006
   -0.3720
    0.1514
    0.0321
    0.0319
    1.6366
fval1 =
  1.0e-006 *
   -0.0058
   -0.0714
   -0.1724
   -0.4240
   -0.3485
   -0.0175
   -0.2495
    0.0705
    0.2478
   -0.0357
exitflag1 =
     1
```

```
options = optimset('Display','iter', 'MaxFunEvals', 3000, 'MaxIter', 1000,
'Jacobian', 'on');
tic; [x2, fval2, exitflag2] = fsolve(@big_matr_jac, x0, options); toc;
```

```
                                        Norm of      First-order   Trust-region
 Iteration  Func-count     f(x)          step         optimality    radius
     0          1         143.886                        52.9              1
     1          2         54.0687             1          15.8              1
     2          3         24.3459           2.5            30            2.5
     3          4        0.546281      0.939948          1.16            2.5
     4          5       0.0740997      0.774046         0.446            2.5
     5          6       0.0020025      0.282759        0.0328            2.5
     6          7     0.000113752      0.159895       0.00688            2.5
     7          8    3.57928e-006      0.0713175        0.0012            2.5
     8          9    1.56366e-008      0.0186663     8.92e-005            2.5
     9         10    4.66329e-013     0.00138452     5.01e-007            2.5
```

```
Elapsed time is 4.871241 seconds.
x2 =
    0.0911
    0.1045
    0.4188
   -0.5588
    0.2006
   -0.3720
    0.1514
    0.0321
    0.0319
    1.6366
fval2 =
  1.0e-006 *
   -0.0058
   -0.0714
   -0.1724
   -0.4240
   -0.3485
   -0.0175
   -0.2495
    0.0705
    0.2478
   -0.0357
>> exitflag2
exitflag2 =
     1
```

Решение системы размерностью 20:

```
N = 20;
generate_data(N);
x0 = randn(N, 1);
options = optimset('Display','iter', 'MaxFunEvals', 3000, 'MaxIter', 1000);
tic; [x1, fval1, exitflag1] = fsolve(@big_matr, x0, options); toc;
```

Iteration	Func-count	f(x)	step	Norm of optimality	First-order radius	Trust-region
0	21	321.872		116		1
1	42	138.26	1	37.5		1
2	63	23.3671	2.5	7.74		2.5
3	64	23.3671	6.03486	7.74		6.25
4	85	11.8665	1.50871	7.74		1.51
5	86	11.8665	1.50871	7.74		1.51
6	107	7.02999	0.377179	3.79		0.377
7	128	2.99173	0.942947	4.04		0.943

```
     8        149       0.0446801       0.731702          0.318           2.36
     9        170      0.00025199       0.217789         0.0287           2.36
    10        191    1.68027e-009      0.0107477      7.23e-005           2.36
    11        212    4.82957e-018   6.46473e-005      3.16e-009           2.36
Elapsed time is 240.786008 seconds.
x1 =
   -0.0665
    0.1092
    0.0457
    0.1972
   -0.1219
   -0.2725
    0.3377
   -0.3037
    0.7025
    0.2005
   -0.2724
   -0.3434
   -0.1833
    0.0642
    0.4835
    0.0419
   -0.1366
    0.4083
    0.2438
    1.2401
fval1 =
  1.0e-008 *
    0.0035
   -0.0043
   -0.0003
   -0.0015
   -0.0008
    0.0290
    0.0398
   -0.0403
    0.1150
   -0.0036
    0.0155
   -0.0361
    0.0113
   -0.0246
    0.0966
```

```
   -0.0291
   -0.0120
    0.1100
    0.0732
   -0.0305
exitflag1 =
     1

options = optimset('Display','iter', 'MaxFunEvals', 3000, 'MaxIter', 1000,
'Jacobian', 'on');
tic; [x2, fval2, exitflag2] = fsolve(@big_matr_jac, x0, options); toc;

                                        Norm of      First-order   Trust-region
 Iteration  Func-count     f(x)          step         optimality    radius
     0          1         321.872                          116             1
     1          2          138.26             1          37.5             1
     2          3         23.3671           2.5          7.74           2.5
     3          4         23.3671       6.03486          7.74          6.25
     4          5         11.8665       1.50871          7.74          1.51
     5          6         11.8665       1.50871          7.74          1.51
     6          7            7.03      0.377179          3.79         0.377
     7          8         2.99174      0.942947          4.04         0.943
     8          9       0.0446804      0.731703         0.318          2.36
     9         10     0.000251992      0.217789        0.0287          2.36
    10         11    1.68034e-009     0.0107478     7.24e-005          2.36
    11         12    4.82605e-018  6.46485e-005     3.16e-009          2.36
x2 =
   -0.0665
    0.1092
    0.0457
    0.1972
   -0.1219
   -0.2725
    0.3377
   -0.3037
    0.7025
    0.2005
   -0.2724
   -0.3434
   -0.1833
    0.0642
    0.4835
    0.0419
   -0.1366
    0.4083
    0.2438
```

```
    1.2401
fval2 =
  1.0e-008 *
    0.0035
   -0.0043
   -0.0003
   -0.0016
   -0.0008
    0.0291
    0.0398
   -0.0403
    0.1149
   -0.0036
    0.0155
   -0.0361
    0.0113
   -0.0246
    0.0965
   -0.0290
   -0.0121
    0.1099
    0.0732
   -0.0305
exitflag2 =

     1
```

Как видно, время решения для системы размерностью 10 составляет 29.95 и 4.87 секунды без якобиана и с якобианом соответственно. Для размерности 20 — 240.78 и 22.89 секунд соответственно. Получаем разницу в первом случае в 6.14 раз, во втором — в 10.5 раз.