Optimization Theory and Algorithms

Primal-dual interior-point method for LP

Assignment:

Write a Matlab function to implement a primal-dual interior-point method (the basic version) as is outlined in the algorithm description, for solving the following linear programming problem

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & Ax = b \\ & x \geq 0 \end{aligned}$$

The header of the function should be as follows:

```
function [x,y,z,iter] = my_pdipm(A,b,c,tol,maxit,prt)
```

where

```
% INPUT:
%               A = constraint coefficient matrix
%               b = constraint right-hand side vector
%               c = objective coefficient vector
%             tol = tolerance
%           maxit = maximum number of iterations allowed
%             prt = switch for screen printout (1 on, 0 off)
% OUTPUT:
%               x = computed primal solution
%               y = computed dual solution
%               z = computed dual slacks
%            iter = iteration counter
```

Procedure:

1. Write your interior-point function according to the specification. The stopping criterion is that either the `maxit` is reached or
$$\frac{\|Ax-b\|}{1+\|b\|} + \frac{\|A^T y + z - c\|}{1+\|c\|} + \frac{|c^T x - b^T y|}{1+|b^T y|} \leq \text{tol}.$$

2. when the switch is on (`prt = 1`), print out the following information on every iteration: iteration count, relative primal and dual residual norms and the relative duality gap, respectively,
$$\frac{\|Ax-b\|}{1+\|b\|}, \quad \frac{\|A^T y + z - c\|}{1+\|c\|}, \quad \frac{|c^T x - b^T y|}{1+|b^T y|}$$
in the format shown below

```
iter  1:  [primal dual gap] = [9.69e+02 1.61e+03 5.13e+05]
iter  2:  [primal dual gap] = [4.43e+02 8.63e+02 5.17e+00]
                               o
                               o
                               o
iter 32:  [primal dual gap] = [6.84e-12 7.79e-17 3.10e-08]
iter 33:  [primal dual gap] = [5.18e-11 7.86e-17 2.21e-11]
Optimization terminated.
```

3. Retrieve the test scripts and instructor's code from the package `handout_pdipm.zip`.

4. For debugging purpose, run the script `test_ipm1` with the default value $r = 1$. Also try the script `test_ipm2(1)` as well.

5. After debugging is done, run `test_ipm1` with $r = 3$ (or $r = 4$ if your code and computer are capable). Also run `test_ipm2(1)` and try `test_ipm2(2)` (if your code can handle it).

6. Submit your codes and the outputs from above runs. Also write a typed, brief writeup to describe points of importance about implementation and performance.