CE3-08 – Process Model Solution and Optimization

Problem Sheet for Topic #2, with Answers

1. Check whether the following functions are convex or concave over the domains specified using the Hessian test. Try to plot the functions in Matlab to validate your answers.

   (a) $f(x) = x\sin(x)$, for $x \in [0, \pi]$? for $x \in [\frac{\pi}{2}, \pi]$?

Answers. The second derivative of f is easily calculated as:

$$f''(x) = 2\cos(x) - x\sin(x)$$

- In the case that $x \in [0, \pi]$, it is found that $f''(0) = 2$ and $f''(\pi) = -2$, thereby showing that f is neither convex nor concave on $[0, \pi]$ (since the curvature of the function is changing sign).
- In the case that $x \in [\frac{\pi}{2}, \pi]$, the following inequalities are easily established:

$$-2 \leq 2\cos(x) \leq 0, \qquad -\pi \leq -x\sin(x) \leq 0,$$

  from where we can conclude that $f''(x) \leq 0$. In other words, f is concave on $[\frac{\pi}{2}, \pi]$.

These results are supported by the plot shown in Fig. 1.

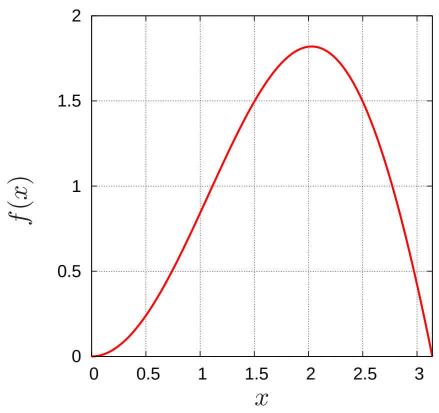


Figure 1: Function f in Question 1a

   (b) $f(x_1, x_2) = 1000 + 7x_1 - 700x_2 + 23x_1x_2 + 10x_1^2 + 5.6x_2^2$, for $x_1, x_2 \in \mathbb{R}$?

Answers. The Hessian matrix of f is given by:

$$\mathbf{H}(\mathbf{x}) = \begin{pmatrix} 20 & 23 \\ 23 & 11.2 \end{pmatrix}$$

Notice that $\mathbf{H}$ is independent of the current point $\mathbf{x}$ since f is a second-order polynomial in x_1, x_2. It can be easily checked whether or not $\mathbf{H}$ is a definite matrix, e.g., by calculating its eigenvalues with Matlab:

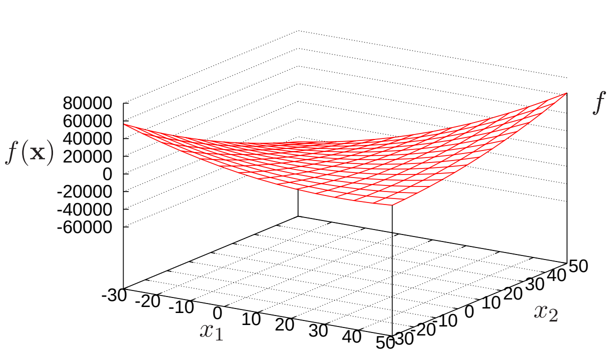


Figure 2: Function f in Question 1b

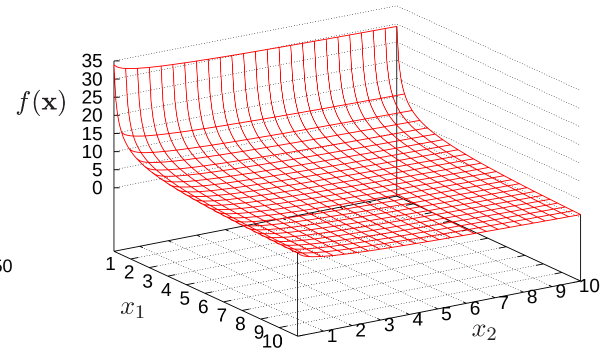


Figure 3: Function f in Question 1c

```
>> eig([[20,23];[23,11.2]])

ans =

   -7.8171
   39.0171
```

Clearly, $\mathbf{H}$ is indefinite, so f is neither convex nor concave. This behavior is confirmed by the plot shown in Fig. 2.

(c) $f(x_1, x_2) = \dfrac{10}{\sqrt{x_1}} - \log(x_2)$, for $x_1, x_2 > 0$?

Answers. Notice that f is the sum of 2 univariate functions in x_1 and x_2, respectively. Accordingly, the convexity/concavity of f is determined by the convexity/concavity of these two terms.

- The function $g(z) = \frac{1}{\sqrt{x}}$ is convex since its 2nd-order derivative $g''(x) = \frac{3}{4z^{\frac{5}{2}}}$ is positive for each $z \in (0, +\infty)$.
- Likewise, the function $h(z) = -\log(z)$ is convex since its 2nd-order derivative $h''(x) = \frac{1}{z^2}$ is positive for each $z \in (0, +\infty)$.

Since the sum of two convex functions yield a convex function, f is therefore convex on $(0, +\infty) \times (0, +\infty)$. The plot in Fig. 3 confirms this analysis.

---

2. For each of the following functions, determine *analytically* whether the specified $\mathbf{x}$ is: (i) definitely a local minimum; (ii) possibility a local minimum; (iii) definitely a local maximum; and/or (iv) possible a local maximum. Try to validate your answers graphically.

(a) $f(x_1, x_2) = 12x_2 - (x_1)^2 + 3x_1x_2 - 3(x_2)^2$, at $\mathbf{x} = (12, 8)$?

Answers. The gradient and the Hessian of f at $\mathbf{x} = (12, 8)$ are easily calculated as:

$$\boldsymbol{\nabla} f(\mathbf{x}) = \begin{pmatrix} -2x_1 + 3x_2 \\ 12 + 3x_1 - 6x_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \qquad \mathbf{H}(\mathbf{x}) = \begin{pmatrix} -2 & 3 \\ 3 & -6 \end{pmatrix}.$$

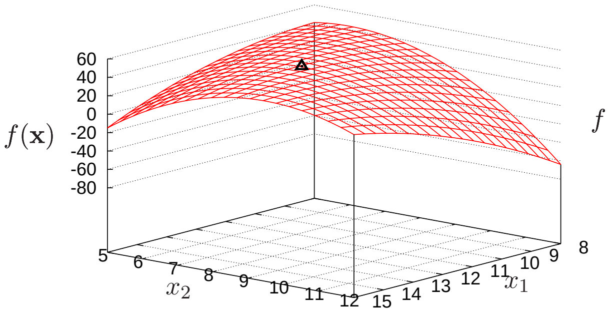


Figure 4: Function f in Question 2a

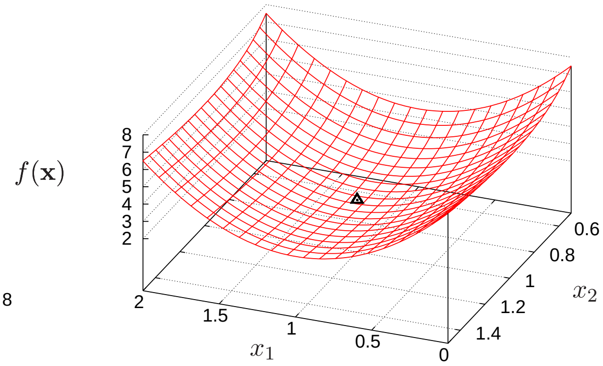


Figure 5: Function f in Question 2b

Clearly, $(12, 8)$ is a stationary point. Moreover, it can be checked that $\mathbf{H}(\mathbf{x})$ is a definite negative matrix using MatLab:

```
>> eig([[-2,3];[3,-6]])

ans =

   -7.6056
   -0.3944
```

One can thus conclude that $(12, 8)$ is a strict local maximum based on the second-order sufficient conditions for optimality. This is illustrated in Fig. 4.

(b) $f(x_1, x_2) = 4(x_1)^2 + \dfrac{3}{x_2} - 8x_1 + 3x_2$, at $\mathbf{x} = (1, 1)$?

Answers. The gradient and the Hessian of f at $\mathbf{x} = (1, 1)$ are given by:

$$\nabla f(\mathbf{x}) = \begin{pmatrix} 8x_1 - 8 \\ -\frac{3}{x_2^2} + 3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \qquad \mathbf{H}(\mathbf{x}) = \begin{pmatrix} 8 & 0 \\ 0 & \frac{6}{x_2^3} \end{pmatrix} = \begin{pmatrix} 8 & 0 \\ 0 & 6 \end{pmatrix}.$$

Clearly, $(1, 1)$ is a stationary point, and it is immediately verified that the diagonal matrix $\mathbf{H}(\mathbf{x})$ is definite positive. One can then invoke the second-order sufficient conditions for optimality to conclude that $(1, 1)$ is a strict local minimum. This is illustrated in Fig. 5.

(c) $f(x_1, x_2) = (x_1)^3(x_2)^2$, at $\mathbf{x} = (0, 0)$?

Answers. The gradient and the Hessian of f at $\mathbf{x} = (0, 0)$ are given by:

$$\nabla f(\mathbf{x}) = \begin{pmatrix} 3x_1^2 x_2^2 \\ 2x_1^3 x_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \qquad \mathbf{H}(\mathbf{x}) = \begin{pmatrix} 6x_1 x_2^2 & 6x_1^2 x_2 \\ 6x_1^2 x_2 & 2x_1^3 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}.$$

Here again, $(0, 0)$ is a stationary point, yet the Hessian matrix is equal to zero at that point. According to the second-order *necessary* conditions for optimality, $(0, 0)$ could be either a local minimum, or a local maximum, or a saddle point. Unfortunately, the sufficient second-order conditions do not apply here, so none of these scenarios can be excluded. By visual inspection, it appears that $(0, 0)$ is a saddle point (see Fig. 6).

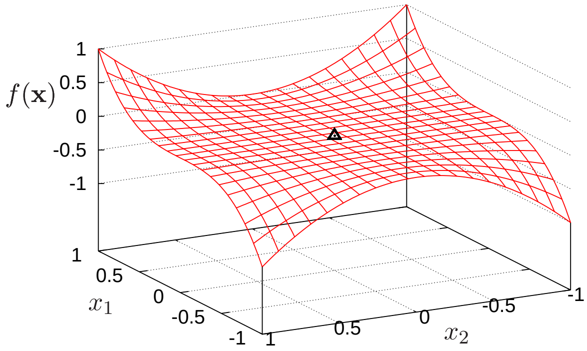
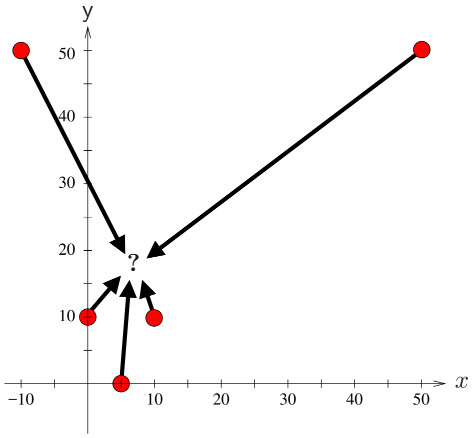


Figure 6: Function f in Question 2c

---

3. We want to connect five oil wells to a single collection point using the minimum total length of pipe, as shown in the figure below. Each pipe segment will be placed in straight line from the well to the collection point.



The locations of the oil wells in the (x, y)-plane are the following:

- $(x_1, y_1) = (5, 0)$
- $(x_2, y_2) = (0, 10)$
- $(x_3, y_3) = (10, 10)$
- $(x_4, y_4) = (50, 50)$
- $(x_5, y_5) = (-10, 50)$

(a) Formulate an (unconstrained) NLP model that can be used to determine where the collection point should be located in the plane.

Answers. The decision variables x and y correspond to the position of the collection point. The function to minimize is:

$$\begin{aligned}\min_{x,y}\; f(x,y) :=& \sum_{k=1}^{5} \sqrt{(x-x_k)^2 + (y-y_k)^2} \\ =& \sqrt{(x-5)^2 + y^2} + \sqrt{x^2 + (y-10)^2} + \sqrt{(x-10)^2 + (y-10)^2} \\ &+ \sqrt{(x-50)^2 + (y-50)^2} + \sqrt{(x+10)^2 + (y-50)^2}\end{aligned}$$

This function is shown in Fig. 7 below.

(b) Could this problem exhibit multiple local optima? Justify your answer.

Answers. It can be shown that every norm in $\mathbb{R}^n$ is a strictly convex function on $\mathbb{R}^n$. In particular, the Euclidean norm defined as $\|\mathbf{x}\|_2 = \sqrt{x_1^2 + x_2^2}$ is convex. Since the

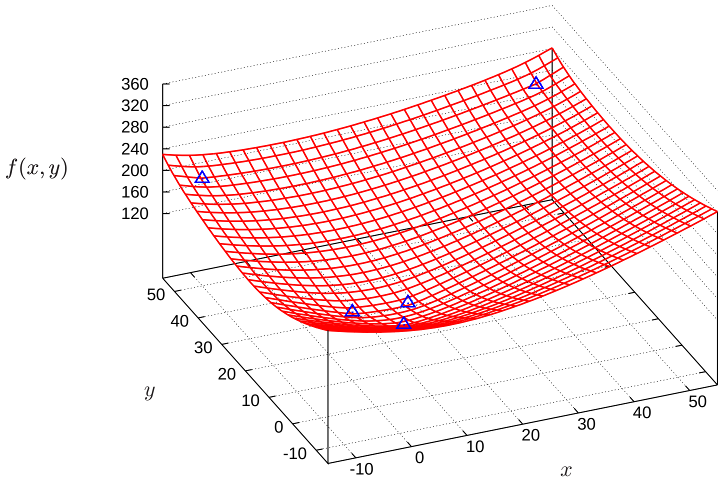


Figure 7: Objective function in Question 3a.

summation of convex functions preserves (strict) convexity, the objective function f is strictly convex on $\mathbb{R}^2$. By convexity, it follows that the optimization problem cannot have local optima that are not global optima, and by strict convexity, that the global optimum is unique.

(c) Implement this problem in Matlab and solve it using the fminunc function. In particular:

- try to provide the gradient of the objective function (see option GradObj)
- set the termination tolerances on the variables and the objective gradient value to 1e-7 (options TolX and TolFun)
- select a 'good' initial guess (e.g., the origin $(0,0)$)

Report the optimal point and solution value found by fminunc, as well as the termination flag, total number of iterations, and measure of first-order optimality. Plot the objective function and compare the actual maximum with the numerical solution.

Answers. A possible implementation based on fminunc is as follows:

oilWells.m

```
% Set options (gradient + tolerances)
Options = optimoptions( @fminunc, 'GradObj', 'on', 'TolFun', 1e-7, 'TolX', 1e-7 );

% Initial guess
p0 = [0; 0];

% Call fminunc
[popt, fopt, Flag, Info] = fminunc( @oilWells_obj, p0, Options )
```

oilWells_obj.m

```
function [ f, df ] = oilWells_obj( p )
% This function calculates the value (f) and first derivative (df)
% of the objective function for a given collection point p

```

```
% Well (x,y) locations
w1 = [   5;  0 ];
w2 = [   0; 10 ];
w3 = [  10; 10 ];
w4 = [  50; 50 ];
w5 = [ -10; 50 ];

% Total length of pipe
f  = norm(p-w1) + norm(p-w2) + norm(p-w3) + norm(p-w4) + norm(p-w5);
df = (p-w1)/norm(p-w1) + (p-w2)/norm(p-w2) + (p-w3)/norm(p-w3) + ...
     (p-w4)/norm(p-w4) + (p-w5)/norm(p-w5);

end
```

The following results are obtained on calling the m-file `oilWells.m` from the command line in Matlab:

Command Line

```
>> oilWells

Local minimum found.

Optimization completed because the size of the gradient is less than
the selected value of the function tolerance.

popt =

   7.618835813588754
  11.214511402349354


fopt =

     1.219537866491946e+02


Flag =

     1


Info =

         iterations: 8
          funcCount: 9
       cgiterations: 8
      firstorderopt: 2.233649931682180e-10
          algorithm: 'large-scale: trust-region Newton'
            message: [1x498 char]
    constrviolation: []
```

These results indicate that an optimum has been found (Flag=1), after 8 iterations. The optimal collection point is $\mathbf{p}^* \approx (7.6188, 11.2145)$ and the corresponding minimal total length of pipe $f^* \approx 121.9538$. Moreover, a measure of first-order optimality by the solver is 2.2×10^{-10}, thereby confirming that the algorithm has converged to a stationary point.

A plot of the optimal solution is shown in Fig. 8 below.

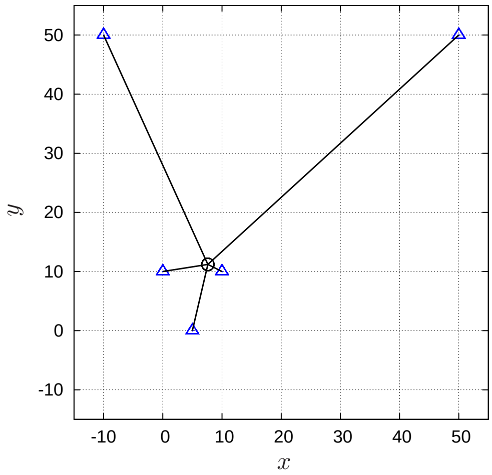


Figure 8: Optimal solution in Question 3c.

(d) Repeat Question 3c using GAMS instead of MATLAB.

Answers. A possible GAMS implementation is as follows:

oilWells.gms

```
SETS
  w  wells / W1*W5 /

PARAMETERS
  x(w)  x-coordinate of wells
   / W1    5
     W2    0
     W3   10
     W4   50
     W5  -10 /
  y(w)  y-coordinate of wells
   / W1    0
     W2   10
     W3   10
     W4   50
     W5   50 /;

VARIABLES
  xc  x-coordinate of collection point
  yc  y-coordinate of collection point
  l   total length of pipe;

EQUATIONS
  length    objective function;

length..    l =E= SUM(w,SQRT(POWER(x(w)-xc,2)+POWER(y(w)-yc,2)));

MODEL collector /ALL/;
SOLVE collector USING NLP MINIMIZING l;
```

The following results can be found in the GAMS report:

oilWells.lst

```
 EXIT - Optimal Solution found, objective:        121.9538

                          LOWER          LEVEL          UPPER         MARGINAL

---- EQU length             .              .              .             1.0000
  length  objective function

                          LOWER          LEVEL          UPPER         MARGINAL

---- VAR xc               -INF            7.6188        +INF      1.0046939E-8
---- VAR yc               -INF           11.2145        +INF      -1.775206E-8
---- VAR l                -INF          121.9538        +INF             .
  xc  x-coordinate of collection point
  yc  y-coordinate of collection point
  l  total length of pipe

**** REPORT SUMMARY :        0     NONOPT
                             0 INFEASIBLE
                             0  UNBOUNDED
                             0     ERRORS

EXECUTION TIME       =        0.000 SECONDS      2 Mb  LEX237-237 Aug 23, 2011
```

These results indicate that an optimum has been found. The optimal collection point is $\mathbf{p}^* \approx (7.6188, 11.2145)$ and the corresponding minimal total length of pipe $f^* \approx 121.9538$ is identical to the solution found previously with MATLAB.

(e) Check that the first- and second-order necessary conditions for optimality are satisfied at the optimum point found previously.

Answers. The gradient and the Hessian of the objective function f are given by:

$$\nabla f(x,y) = \begin{pmatrix} \displaystyle\sum_{k=1}^{5} \frac{x - x_k}{\sqrt{(x-x_k)^2 + (y-y_k)^2}} \\ \displaystyle\sum_{k=1}^{5} \frac{y - y_k}{\sqrt{(x-x_k)^2 + (y-y_k)^2}} \end{pmatrix}$$

$$\mathbf{H}(x,y) = \begin{pmatrix} \displaystyle\sum_{k=1}^{5} \frac{(y-y_k)^2}{((x-x_k)^2 + (y-y_k)^2)^{\frac{3}{2}}} & \displaystyle\sum_{k=1}^{5} \frac{(x-x_k)(y-y_k)}{((x-x_k)^2 + (y-y_k)^2)^{\frac{3}{2}}} \\ \displaystyle\sum_{k=1}^{5} \frac{(x-x_k)(y-y_k)}{((x-x_k)^2 + (y-y_k)^2)^{\frac{3}{2}}} & \displaystyle\sum_{k=1}^{5} \frac{(x-x_k)^2}{((x-x_k)^2 + (y-y_k)^2)^{\frac{3}{2}}} \end{pmatrix}$$

The second-order derivatives can be added to the m-file `oilWells_obj.m` as follows:

oilWells_obj.m

```
function [ f, df,d2f ] = oilWells_obj( p )
% This function calculates the value (f), first derivative (df), and
```

```
% second derivative (d2f) of the objective function for a given
% collecion point p

% Well (x,y) locations
w1 = [   5;  0 ];
w2 = [   0; 10 ];
w3 = [  10; 10 ];
w4 = [  50; 50 ];
w5 = [ -10; 50 ];

% Total length of pipe
f  = norm(p-w1) + norm(p-w2) + norm(p-w3) + norm(p-w4) + norm(p-w5);
df = (p-w1)/norm(p-w1) + (p-w2)/norm(p-w2) + (p-w3)/norm(p-w3) + ...
     (p-w4)/norm(p-w4) + (p-w5)/norm(p-w5);
Q = ones(2)-eye(2); % Permutation matrix
d2f = Q* ( (p-w1)*(p-w1)'/norm(p-w1)^3 + (p-w2)*(p-w2)'/norm(p-w2)^3 + ...
           (p-w3)*(p-w3)'/norm(p-w3)^3 + (p-w4)*(p-w4)'/norm(p-w4)^3 + ...
           (p-w5)*(p-w5)'/norm(p-w5)^3 ) * Q';

end
```

The following results are obtained upon calling the m-file `oilWells_obj.m` with the optimum point $\mathbf{p}^*$ found in Question 3c:

Command Line

```
>> [fopt,dfopt,d2fopt] = oilWells_obj( popt )

fopt =

  121.9538


dfopt =

   1.0e-09 *

    0.2067
    0.2234


d2fopt =

    0.1902   -0.1122
   -0.1122    0.4413

>> eig( d2fopt )

ans =

    0.1473
    0.4841
```

It is checked that the gradient elements are both very close to zero. Moreover, both eigenvalues of the Hessian matrix are positive, thereby confirming that $\mathbf{p}^*$ is indeed a strict local minimum – in fact, a strict global minimum by convexity.

4. Consider the following QP problem:

$$\min_{\mathbf{x}} \; x_1^2 + x_2^2 - 8x_1 - 16x_2 + 32$$
$$\text{s.t. } x_1 + x_2 \leq 5$$
$$x_1 \geq 0$$

(a) Write down the KKT conditions for this QP. Identify such KKT points, by successively considering *all* possible active sets (4 possibilities).

Answers. The conditions for $(\mathbf{x}, \boldsymbol{\nu})$ to be a KKT point for this problem are the following:

$$\begin{aligned} x_1 + x_2 &\leq 5, & \nu_1(x_1 + x_2 - 5) &= 0, & \nu_1 &\geq 0,\\ -x_1 &\leq 0, & \nu_2 x_1 &= 0, & \nu_2 &\geq 0,\\ 2x_1 - 8 + \nu_1 - \nu_2 &= 0, & 2x_2 - 16 + \nu_1 &= 0, \end{aligned}$$

Next, we search KKT points for different active-set configuration:

Constraints #1 & #2 Inactive: This case corresponds to $\nu_1 = \nu_2 = 0$. The dual feasibility conditions therefore reduce to the system:

$$2x_1 - 8 = 0, \qquad 2x_2 - 16 = 0,$$

the unique solution of which is $x_1 = 4$ and $x_2 = 8$. However, this point violates the first inequality constraint since $4 + 8 = 12 > 5$. Therefore, a KKT point cannot exist in this active-set configuration.

Constraint #1 Active & #2 Inactive: This case corresponds to $\nu_2 = 0$, and $x_1 + x_2 = 5$. This later condition along with the dual feasibility conditions yield the following (linear) system:

$$x_1 + x_2 = 5, \qquad 2x_1 - 8 + \nu_1 = 0, \qquad 2x_2 - 16 + \nu_1 = 0,$$

the unique solution of which is $x_1 = \frac{1}{2}$, $x_2 = \frac{9}{2}$ and $\nu_1 = 7$. Note that $\nu_1 = 7 \geq 0$ also satisfies the sign restriction of the dual feasibility conditions. All the KKT conditions being satisfied, we have thus found a KKT point.

Constraint #1 Inactive & #2 Active: This case corresponds to $\nu_1 = 0$, and $x_1 = 0$. This later condition along with the dual feasibility conditions yield the following (linear) system:

$$x_1 = 0, \qquad 2x_1 - 8 - \nu_2 = 0, \qquad 2x_2 - 16 = 0,$$

the unique solution of which is $x_1 = 0$, $x_2 = 8$ and $\nu_2 = -8$. Note that $\nu_2 = -8 < 0$ violates the sign condition imposed by the dual feasibility conditions. Therefore, a KKT point cannot exist in this active-set configuration.

Constraints #1 & #2 Active: This case corresponds to $x_1 + x_2 = 5$, and $x_1 = 0$, i.e., $x_2 = 5$. The dual feasibility conditions therefore reduce to the system:

$$-8 + \nu_1 - \nu_2 = 0, \qquad\qquad -6 + \nu_1 = 0,$$

the unique solution of which is $\nu_1 = 6$ and $\nu_2 = -2$. Here again, $\nu_2 = -2 < 0$ violates the sign condition imposed by the dual feasibility, so no KKT point can exist in this active-set configuration.

Overall, the unique KKT point for this NLP problem is:

$$x_1^* = \frac{1}{2}, \qquad x_2^* = \frac{9}{2}, \qquad \nu_1^* = 7, \qquad \nu_2^* = 0$$

(b) Check whether the QP is convex. Conclude.

Answers. A convex optimization problem involves the minimization of a convex objective function over a convex region. A convex domain is obtained when (i) *all* the functions participating in constraints of ≤ 0 type are convex on the search space, and those participating in constraints of ≥ 0 type are concave; and (ii) *all* the functions participating in constraints of $= 0$ type are affine.
In the problem at hand, both inequality constraints are affine, and therefore define a convex domain. Moreover, the objective function is a quadratic polynomial in the variables x_1, x_2; its Hessian matrix,

$$\mathbf{H}(\mathbf{x}) := \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix},$$

being positive definite, this function is (strictly) convex. Overall, this optimization problem is therefore a convex QP.
From the first-order sufficient conditions of optimality, it follows that any KKT point yields a *global* optimum for that problem (no matter of the constraints being regular or not). The KKT point determined previously in Question 4a, $\mathbf{x}^* := \begin{pmatrix} \frac{1}{2} & \frac{9}{2} \end{pmatrix}^{\mathsf{T}}$, is therefore the unique global optimum for our problem.

---

5. In this question, you are to develop and test an implementation of the active-set method for QP optimization, as described on Slide #63 of Lecture 3. Your function will comply with the following:

qp.m

```
function [xopt, fopt] = qp( c, Q, Ae, be, Ai, bi )

  % <- TO BE COMPLETED

end
```

The input arguments of this function are as follows:

- `c` , the linear part of the objective function

- Q, the quadratic part of the objective function
- Ae, the coefficient matrix of equality constraints
- be, the right-hand-side vector of equality constraints
- Ai, the coefficient matrix of inequality constraints
- bi, the right-hand-side vector of inequality constraints

On successful completion, xopt and fopt should contain, respectively, an (approximate) optimum point and the corresponding solution value.

(a) Develop the m-file qp.m in order to implement the implementing to active-set method, in the case of a minimization problem. Make sure to test that the QP is convex inside the function.

Answers. A possible implementation of the active-set QP algorithm is given below. In addition to returning the optimal point and solution value of the QP, we have defined an extra output argument returning the optimal multipliers for the linear equality and inequality constraints here.

qp.m

```
function [xopt, fopt, lopt] = qp( c, Q, Ae, be, Ai, bi )

% Check positive semi-definiteness
if( eigs( Q, 1, 'SA' ) < 0 )
   error('Objective function is nonconvex');
end

% Initialization
n  = length( c );
me = length( be );
mi = length( bi );
act = []; % start with no active inequality constraints

% Main loop - repeat until active set does not change anymore
cnt = true;
while( cnt )
   % Check feasibility
   ma = length( act );
   if( me+ma > n )
      error('Infeasible problem');
   end

   % Solve equality constrained QP with current active set
   if( me )
     M = [ [ Q         Ae'  Ai(act,:)' ]
           [ Ae        zeros(me,me+ma) ]
           [ Ai(act,:) zeros(ma,me+ma) ] ];
     N = [ -c; be; bi(act) ];
   else
     M = [ [ Q         Ai(act,:)' ]
           [ Ai(act,:) zeros(ma,ma) ] ];
     N = [ -c; bi(act) ];
   end
   y = M \ N;
   cnt = false;

```

```
    % Remove inequality constraints with negative multipliers
    for i = 1:ma
       if( y(n+me+i) < 0. )
          cnt = true;
          act = [ act(1:n+me+i-1) act(n+me+i+1:end) ];
       end
    end

    % Append violated inequality constraints
    for j = 1:mi
       if( Ai(j,:)*y(1:n) > bi(j) )
          cnt = true;
          act = [ act j ];
       end
    end
end

% Gather results
xopt = y(1:n);
lopt = struct('eqlin',y(n+1:n+me),'ineqlin',zeros(mi,1));
lopt.ineqlin(act) = y(n+me+1:end);
fopt = c'*xopt + 0.5*xopt'*Q*xopt;

end
```

(b) Debug and test your m-file by considering the same problem as in Question 4 above.

Answers. The following results are obtained upon calling the m-file `qp.m` to solve the QP in Question 4:

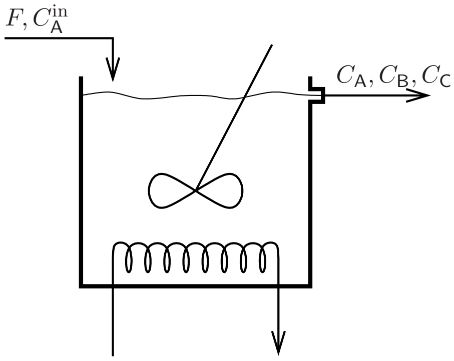
Command Line

```
>> [popt, fopt, lopt] = qp( [-8;-16], 2.*eye(2), [], [], [[1 1];[-1 0]], [5;0] )

popt =

    0.5000
    4.5000


fopt =

  -55.5000


lopt =

      eqlin: [0x1 double]
    ineqlin: [2x1 double]

>> lopt.ineqlin

ans =

     7
     0
```

Note that the numerical values match the global optimum determined analytically in Question 4.

---

6. Consider the isothermal, constant volume CSTR with series reaction A $\xrightarrow{k_1}$ B $\xrightarrow{k_2}$ C shown in the figure below. Our goal is to maximize the concentration of species B, C_{B}, by adjusting the flow rate F and inlet concentration $C_{\mathsf{A}}^{\text{in}}$ simultaneously, while keeping all other variables and parameters constant (see table below). Both manipulated variables F and $C_{\mathsf{A}}^{\text{in}}$ should be nonnegative and their maximum allowable levels are F_U and $C_{\mathsf{A}}^{\text{in,U}}$, respectively.



Param.	Value	
k_1	5.0	min^{-1}
k_2	2.0	min^{-1}
V	1.0	L
C_{A}^U	0.2	mol L^{-1}
$C_{\mathsf{A}}^{\text{in},U}$	1.0	mol L^{-1}
F^U	20.0	L min^{-1}

Figure 9: CSTR with series reaction and problem data.

(a) This optimization problem is first modeled as the following constrained NLP:

$$\boxed{\begin{aligned} \max_{F, C_{\mathsf{A}}^{\text{in}}} \quad & C_{\mathsf{A}}^{\text{in}} \frac{F k_1 V}{(F + k_1 V)(F + k_2 V)} \\ \text{s.t.} \quad & C_{\mathsf{A}}^{\text{in}} \frac{F}{F + k_1 V} \leq C_{\mathsf{A}}^U \\ & 0 \leq F \leq F^U \\ & 0 \leq C_{\mathsf{A}}^{\text{in}} \leq C_{\mathsf{A}}^{\text{in},U} \end{aligned}}$$

i. Implement this optimization problem in Matlab, then solve it using the fmincon function, starting from the initial values $F = 10\ \text{L min}^{-1}$, $C_{\mathsf{A}}^{\text{in}} = 0.5\ \text{mol L}^{-1}$. Check that the solver terminated successfully. Report the optimal point and solution value found by fmincon, as well as the optimal active set. Comment.

Indications. Use finite difference approximations for the gradients of the objective and constraint functions for simplicity (see options GradObj and GradCon); set the termination tolerances on the variables, constraints and KKT conditions value to 1e-7 (options TolX TolCtr and TolFun).

Answers. A possible implementation based on fmincon is as follows:

CSTR.m

```
% Parameters
global k1 k2 V CAU
k1    = 5.;  % kinetic constant (min-1)
k2    = 2.;  % kinetic constant (min-1)
V     = 1.;  % reactor volume (L)
CAU   = 0.2; % max. outlet conc. of A (mol L-1)
```

```
CAinU = 1.;  % max. inlet conc. of A (mol L-1)
FU    = 20.; % max. inlet flow rate (L min-1)

% Set options (gradient + tolerances)
Options = optimoptions( @fmincon, 'GradObj', 'off', 'GradConstr', 'off', ...
    'TolFun', 1e-7, 'TolCon', 1e-7, 'TolX', 1e-7, 'display', 'iter', ...
    'algorithm', 'active-set' );

% Variables are: x = [ F  CAin ]
% Initial guess: F = 10 L/min, CAin = 0.5 mol/L
x0 = [ 10; 0.5 ];

% Call fmincon
[xopt, fopt, flag, out, lopt] = fmincon( @(x)CSTR_obj(x), x0, [], [], [], [], ...
    zeros(2,1), [FU; CAinU], @(x)CSTR_ctr(x), Options )
```

CSTR_obj.m

```
function f = CSTR_obj( x )

% Variables are: x = [ F  CAin ]
global k1 k2 V
f = -x(2)*x(1)*k1*V/(x(1)+k1*V)/(x(1)+k2*V);

end
```

CSTR_ctr.m

```
function [ gin, geq ] = CSTR_ctr( x )

% Variables are: x = [ F  CAin ]
global k1 V CAU
gin = [ x(2)*x(1)/(x(1)+k1*V) - CAU ];
geq = [];

end
```

The following results are obtained on calling the m-file `CSTR.m` from the command line in Matlab:

Command Line

```
>> CSTR
                              Max    Line search Directional First-order
 Iter F-count       f(x)  constraint  steplength  derivative  optimality Procedure
    0      3   -0.138889     0.1333                                     Infeasible start point
    1      6  -0.0834645   6.606e-05           1       0.276       0.199
    2      9  -0.0834776  -9.916e-09           1    -0.00224     0.00696  Hessian modified
    3     12  -0.0860956  -9.396e-05           1    -0.00697     0.00741  Hessian modified twice
    4     15    -0.08894  -0.0001065           1    -0.00752     0.00823  Hessian modified twice
    5     18  -0.0921848  -0.0001351           1    -0.00803     0.00928  Hessian modified twice
    6     21  -0.0959395  -0.0001757           1    -0.00865      0.0113  Hessian modified twice
    7     24   -0.100359  -0.0002359           1    -0.00939       0.013  Hessian modified twice
    8     27   -0.105674  -0.0003296           1     -0.0103      0.0158  Hessian modified twice
    9     30   -0.112246  -0.0004853           1     -0.0115       0.019  Hessian modified twice
   10     33   -0.120669  -0.0007662           1     -0.0131      0.0238  Hessian modified twice
   11     36   -0.132019   -0.001336           1     -0.0152      0.0308  Hessian modified twice
   12     39   -0.148411   -0.002706           1     -0.0185      0.0424  Hessian modified twice
   13     42   -0.174402   -0.007032           1     -0.0241      0.0634  Hessian modified twice
   14     45   -0.214993    -0.02953           1     -0.0354       0.102  Hessian modified twice
   15     48   -0.216396           0           1      -0.079      0.0859  Hessian modified twice
   16     51   -0.301523           0           1      -0.214      0.0299
   17     54   -0.307637           0           1       -0.11     0.00243
   18     57   -0.307692           0           1      -0.102    2.81e-06  Hessian modified

Local minimum found that satisfies the constraints.
```

```
Optimization completed because the objective function is non-decreasing in
feasible directions, to within the selected value of the function tolerance,
and constraints are satisfied to within the selected value of the constraint tolerance.

<stopping criteria details>

Active inequalities (to within options.TolCon = 1e-07):
  lower      upper     ineqlin   ineqnonlin
               2                      1

xopt =

    1.2500
    1.0000


fopt =

   -0.3077


flag =

     1


out =

         iterations: 19
          funcCount: 57
       lssteplength: 1
           stepsize: 4.6997e-08
          algorithm: [1x44 char]
      firstorderopt: 4.8054e-09
    constrviolation: 0
            message: [1x787 char]


lopt =

         lower: [2x1 double]
         upper: [2x1 double]
         eqlin: [0x1 double]
      eqnonlin: [0x1 double]
       ineqlin: [0x1 double]
    ineqnonlin: 0.7988

>> lopt.ineqnonlin

ans =

    0.7988

>> lopt.upper

ans =

         0
    0.1479
```

- The optimal solution point for this problem is: $F^* = 1.250\ \mathrm{L\,min^{-1}}$, $C_{\mathsf{A}}^{\mathrm{in}*} = 1.000\ \mathrm{mol\,L^{-1}}$
- The corresponding optimal solution value (cost) is: $C_{\mathsf{B}}^* = 0.308\ \mathrm{mol\,L^{-1}}$ (remember we are minimizing $-C_{\mathsf{B}}$ here)
- The optimal active set consists of the inequality constraints $C_{\mathsf{A}}^{\mathrm{in}} \frac{F}{F+k_1 V} \leq C_{\mathsf{A}}^U$ and $C_{\mathsf{A}}^{\mathrm{in}} \leq C_{\mathsf{A}}^{\mathrm{in},U}$; all other inequality constraints being inactive at the optimum

These results are confirmed by the graphical solution shown in Fig. 10.

ii. Repeat Question 6(a)i using GAMS instead of MATLAB. Comapre the results.

Indications. GAMS allows a user to fix the activity levels (values) of variables through the `.l` suffix:

```
VARIABLE x1;
x1.l = 1.0;
```

assigns the value 1 to the variable `x1`. In particular, values assigned to the variables prior to the `SOLVE` statement serve as initial values for the solver – default values of 0 are used otherwise. This is particularly important for NLP problems. For further information, you

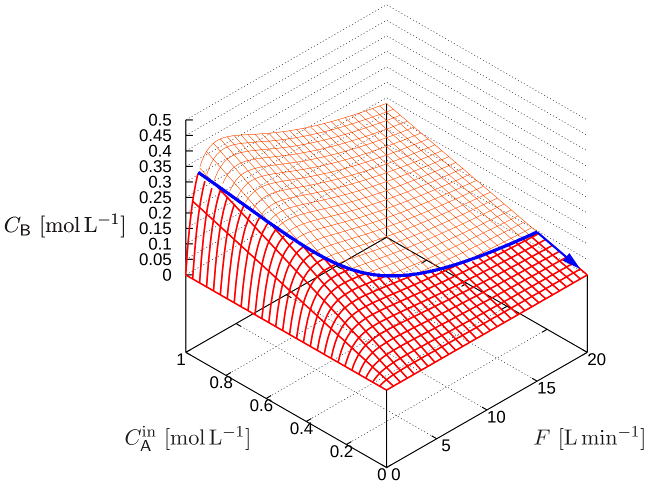


Figure 10: Graphical solution for CSTR with series reaction.

can always refer to the GAMS users guide: `http://www.gams.com/dd/docs/bigdocs/GAMSUsersGuide.pdf`.

Answers. A possible GAMS implementation is as follows:

CSTR.gms

```
SCALARS
 K1     'kinetic constant (min-1)'          / 5.0 /
 K2     'kinetic constant (min-1)'          / 2.0 /
 V      'reactor volume (L)'                / 1.0 /
 CAU    'max. outlet conc. of A (mol L-1)'  / 0.2 /
 CAINU  'max. inlet conc. of A (mol L-1)'   / 1.0 /
 FU     'max. inlet flow rate (L min-1)'    / 20.0 /;

VARIABLES
 F     'inlet flow rate (L min-1)'
 CAIN  'inlet conc. of A (mol L-1)'
 CB    'outlet conc. of B (mol L-1)';

POSITIVE VARIABLES F, CAin;
F.UP    = FU;
CAIN.UP = CAINU;

EQUATIONS
 CBOUT  'objective'
 CAMAX  'residual concentration constraint';

CBOUT..  CB =E= CAIN*F*K1*V/((F+K1*V)*(F+K2*V));
CAMAX..  CAIN*F/(F+K1*V) =L= CAU;

MODEL CSTR / CBOUT, CAMAX /;

F.l    = 10.0;
CAIN.l = 0.5;

SOLVE CSTR USING NLP MAXIMIZING CB;
```

The following results can be found in the GAMS report:

```
                                   CSTR.lst

                          LOWER     LEVEL     UPPER    MARGINAL

---- EQU CBOUT              .         .         .        1.000
---- EQU CAMAX            -INF      0.200     0.200      0.799

  CBOUT  objective
  CAMAX  residual concentration constraint

                          LOWER     LEVEL     UPPER    MARGINAL

---- VAR F                  .        1.250    20.000       .
---- VAR CAIN               .        1.000     1.000     0.148
---- VAR CB               -INF      0.308     +INF        .

  F  inlet flow rate (L min-1)
  CAIN  inlet conc. of A (mol L-1)
  CB  outlet conc. of B (mol L-1)

**** REPORT SUMMARY :        0     NONOPT
                             0 INFEASIBLE
                             0  UNBOUNDED
                             0     ERRORS
```

The optimal point, solution value and active set the same as in Question 6(a)i.

iii. What is the rate of change in optimal concentration C_{B} for a variation in the maximum allowable concentration C_{A}^U? Using this value (and without any additional calculations in GAMS or MATLAB), estimate the change in C_{B} incurred by increasing C_{A}^U to 0.25 mol L^{-1}.

Answers. The rate of change ν^* in optimal concentration C_{B}^* for a variation in the maximum allowable concentration C_{A}^U is given by the marginal value (KKT multiplier) of the constraint $C_{\mathsf{A}}^{\text{in}}\frac{F}{F+k_1V} \leq C_{\mathsf{A}}^U$ (equation CAMAX in the above model):

$$\nu^* \triangleq \frac{\partial C_{\mathsf{B}}^*}{\partial C_{\mathsf{A}}^U} = 0.799 \text{ (mol L}^{-1})/(\text{mol L}^{-1}).$$

Based on this value, an estimate of the change in C_{B} incurred by increasing C_{A}^U to 0.25 mol L^{-1} is obtained as follows:

$$\Delta C_{\mathsf{B}}^* \approx \nu^* \times \Delta C_{\mathsf{A}}^U = 0.799 \times 0.05 = 0.040 \text{ mol L}^{-1}.$$

In other words, an estimate of the cost value for $C_{\mathsf{A}}^U = 0.25$ mol L^{-1} is: $C_{\mathsf{B}}^* \approx 0.348$ mol L^{-1}.

iv. Modify, then resolve, your GAMS or MATLAB model to reflect the previous change in C_{A}^U. Report the optimal solution value, and compare it to the estimate calculated in the previous question. Discuss your results.

Answers. We make the modification on the GAMS model here. The only required change is in line #5:

```
CAU    'max. outlet conc. of A (mol L-1)'  / 0.25 /
```

The following results are obtained on running the modified GAMS model:

CSTR.lst

```
                          LOWER     LEVEL     UPPER    MARGINAL

---- EQU CBOUT              .         .         .        1.000
---- EQU CAMAX            -INF      0.250     0.250     0.537

  CBOUT  objective
  CAMAX  residual concentration constraint

                          LOWER     LEVEL     UPPER    MARGINAL

---- VAR F                  .        1.667    20.000      .
---- VAR CAIN               .        1.000     1.000     0.207
---- VAR CB               -INF      0.341     +INF       .

  F  inlet flow rate (L min-1)
  CAIN  inlet conc. of A (mol L-1)
  CB  outlet conc. of B (mol L-1)

**** REPORT SUMMARY :        0     NONOPT
                             0 INFEASIBLE
                             0  UNBOUNDED
                             0     ERRORS
```

The optimal solution value now is: $C_\mathsf{B}^* = 0.341\ \mathrm{mol\,L^{-1}}$. Observe that this value is rather close to the estimate ($C_\mathsf{B}^* \approx 0.348\ \mathrm{mol\,L^{-1}}$) calculated in the previous question. Observe also that the set of active constraints remains unchanged. However, unlike LP, sensitivity analysis in NLP can only provide an *approximation* of the actual change, due to the nonlinearity.

(b) You notice that the same optimization problem could be modeled by incorporating the mole-balance equations directly as equality constraints, thereby yielding the *equivalent* NLP:

$$
\begin{array}{rl}
\displaystyle\max_{F, C_\mathsf{A}^{\mathrm{in}}, C_\mathsf{A}, C_\mathsf{B}} & C_\mathsf{B} \\
\text{s.t.} & 0 = F\left(C_\mathsf{A}^{\mathrm{in}} - C_\mathsf{A}\right) - k_1 C_\mathsf{A} V \\
& 0 = -F C_\mathsf{B} + \left(k_1 C_\mathsf{A} - k_2 C_\mathsf{B}\right) V \\
& C_\mathsf{A} \le C_\mathsf{A}^U \\
& 0 \le F \le F^U \\
& 0 \le C_\mathsf{A}^{\mathrm{in}} \le C_\mathsf{A}^{\mathrm{in},U}
\end{array}
$$

i. Repeat Question 6(a)ii for this reformulated optimization model.

Answers. A possible GAMS implementation is as follows:

CSTR2.gms

```
SCALARS
 K1     'kinetic constant (min-1)'         / 5.0 /
 K2     'kinetic constant (min-1)'         / 2.0 /
 V      'reactor volume (L)'               / 1.0 /
 CAU    'max. outlet conc. of A (mol L-1)'  / 0.2 /
 CAINU  'max. inlet conc. of A (mol L-1)'   / 1.0 /
```

```
 FU     'max. inlet flow rate (L min-1)'    / 20.0 /;

VARIABLES
 F     'inlet flow rate (L min-1)'
 CAIN  'inlet conc. of A (mol L-1)'
 CA    'outlet conc. of A (mol L-1)'
 CB    'outlet conc. of B (mol L-1)';

POSITIVE VARIABLES F, CAin;
F.UP    = FU;
CAIN.UP = CAINU;
CA.UP   = CAU;

EQUATIONS
 MBALA  'mole balance for species A'
 MBALB  'mole balance for species B';

MBALA..  F*(CAIN-CA) - K1*CA*V   =E= 0;
MBALB..  -F*CB + (K1*CA-K2*CB)*V =E= 0;

MODEL CSTR / MBALA, MBALB /;

F.l    = 10.0;
CAIN.l = 0.5;

SOLVE CSTR USING NLP MAXIMIZING CB;
```

The following results can be found in the GAMS report:

CSTR2.lst

```
                       LOWER     LEVEL     UPPER    MARGINAL

---- EQU MBALA           .         .         .       -0.118
---- EQU MBALB           .         .         .       -0.308

  MBALA  mole balance for species A
  MBALB  mole balance for species B

                       LOWER     LEVEL     UPPER    MARGINAL

---- VAR F               .        1.250    20.000      .
---- VAR CAIN            .        1.000     1.000     0.148
---- VAR CA            -INF      0.200     0.200     0.799
---- VAR CB            -INF      0.308     +INF       .

  F  inlet flow rate (L min-1)
  CAIN  inlet conc. of A (mol L-1)
  CA  outlet conc. of A (mol L-1)
  CB  outlet conc. of B (mol L-1)

**** REPORT SUMMARY :        0     NONOPT
                             0 INFEASIBLE
                             0  UNBOUNDED
                             0     ERRORS
```

- As expected, the optimal solution point for the reformulated problem is identical to the one found previously: $F^* = 1.250\ \mathrm{L\,min^{-1}}$, $C_{\mathsf{A}}^{\mathrm{in}*} = 1.000\ \mathrm{mol\,L^{-1}}$

- The corresponding optimal solution value (cost) is also identical: $C_{\mathsf{B}}^{*} = 0.308\ \mathrm{mol\,L^{-1}}$
- The optimal active set consists of the bound constraints $C_{\mathsf{A}} \leq C_{\mathsf{A}}^{U}$ and $C_{\mathsf{A}}^{\mathrm{in}} \leq C_{\mathsf{A}}^{\mathrm{in},U}$ (with the same marginal values as previously), along with the two equality constraints; all other inequality constraints being inactive at the optimum.