MAE 598: Design Optimisation

Homework 5 Solution

Problem 1

```
clc; clear all;

% Defining functions and constraints
func= @(x)x(1)^2+(x(2)-3)^2;
dfd = @(x)[2*x(1),(x(2)-3)*2];
g = @(x)[x(2)^2-2*x(1);(x(2)-1)^2+5*x(1)-15];
dgd = @(x)[-2 2*x(2); 5 2*(x(2)-1)];

% Declaring parameters
er=1e-3; % Error tolerance
x0=[1;1]; % Initial guess
x=x0;

% Calculating required variables
W=eye(length(x)); % Hessian
mew=zeros(size(g(x))); % μ
wts=zeros(size(g(x))); % Weights of the merit function
lgr=norm(dfd(x)+mew'*dgd(x)); % Norm of the Lagrangian

while lgr>er
    % Initialising the in-built quadratic programming solver
    options = optimoptions('quadprog','Algorithm','active-set');
    [s,~,~,~,lambda] = quadprog(W,dfd(x)',dgd(x),-g(x),[],[],[],[],x,options); % Calculating the s and lambda from solver
    mew1=lambda.ineqlin; % Extracting μ from the output lambda

    % Linesearch steps
    wts = max(abs(mew),0.5*(wts+abs(mew))); % Calculating weights
    alpha=lineSearch(func,dfd,g,dgd,x,s,wts);
    sk=alpha*s; x1=x+sk; % Advancing x

    %BFGS
    dlgrd=[dfd(x1)+mew1'*dgd(x1)-dfd(x)+mew1'*dgd(x)]'; % Gradient of Lagrangian
    if sk'*dlgrd >=0.2*sk'*W*sk
```

```
        theta=1;
    else
        theta=(0.8*sk'*W*sk)/(sk'*W*sk-(sk'*dlgrd));
    end

    y=theta*dlgrd+(1-theta)*W*sk;
    W=W+(y*y')/(y'*sk)-((W*sk)*(sk'*W))/(sk'*W*sk); % Updating Hessian

    mew=mew1;
    x=x1;
    lgr=norm(dfd(x)+mew'*dgd(x));

end

fprintf('The final point after minimisation efforts is calculated to be\n x1 = %f; x2 = %f',x(1),x(2))


% Function for Linesearch Algorithm for step size
function alpha = lineSearch(func,dfd,g,dgd,x,s,w)
    b=0.5; t=0.3; alpha=1;
    fa=func(x+alpha*s)+w'*abs(min(0,-g(x+alpha*s)));
    p1=func(x)+w'*abs(min(0,-g(x)));
    p2=dfd(x)*s+w'*((dgd(x)*s)).*(g(x)>0);
    phi=p1+t*alpha*p2;
    while fa>phi
        alpha=alpha*b;
        fa=func(x+alpha*s)+w'*abs(min(0,-g(x+alpha*s)));
        p1=func(x)+w'*abs(min(0,-g(x)));
        p2=dfd(x)*s+w'*((dgd(x)*s)).*(g(x)>0);
        phi=p1+t*alpha*p2;
    end
end
```

```
Minimum found that satisfies the constraints.

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


Minimum found that satisfies the constraints.
```

```
The problem is non-convex.


Minimum found that satisfies the constraints.

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


Minimum found that satisfies the constraints.

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


Minimum found that satisfies the constraints.

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


The problem is non-convex.


Minimum found that satisfies the constraints.

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

The final point after minimisation efforts is calculated to be
 x1 = 1.060208; x2 = 1.456165
```

Problem 2

Given. states are $[h, v, m]^T$, with dynamics as

$$\dot{h}(t) = v(t)$$
$$\dot{v}(t) = -g + a(t)/m(t)$$
$$\dot{m}(t) = -K a(t)$$

h is the altitude, v is the velocity & m is the mass of the lander $a(t) \in [0, 1]$ is the thrust and K is the const rate of fuel burning

We have initial states as $[h_0, v_0, m_0]^T$ and the final results that we want are $h(t^*) = 0$ & $v(t^*) = 0$ where t^* is the final time.

Our objective is to minimize the fuel consumption, which depends upon if the landing is ~~front~~ firing its engines and thrust is being generated.

$$\min_{\alpha(t)} P(\alpha) = \int_0^{t^*} \alpha(t) \cdot dt$$

So ~~ver~~ our loss function will be, $l = a(t)$ and the dynamics of the system will be,

$$f = \begin{bmatrix} \dot{h}(t) \\ \dot{v}(t) \\ \dot{m}(t) \end{bmatrix} = \begin{bmatrix} v(t) \\ -g + a(t)/m(t) \\ -K a(t) \end{bmatrix}$$

and our Lagrangian multipliers and $\lambda = [\lambda_1\ \lambda_2\ \lambda_3]^T$

So our hamiltonian is $H = -l + \lambda^T f$

$\Rightarrow H = -a(t) + \lambda_1 v(t) + \lambda_2 (-g + a(t)/m(t)) - \lambda_3 k a(t)$

The optimal policy is to maximise the H,

so $a^* = \underset{a \in [0,1]}{\arg\max} H = \underset{a \in [0,1]}{\arg\max} \left(-1 + \frac{\lambda_2}{m} - \lambda_3 k\right) a + \lambda_1 v - \lambda_2 g$

As the Hamiltonian is a linear function w.r.t control, the optimal control policy is bang bang.

so in H, $(-1 + \lambda_2/m - \lambda_3 k) \equiv b$

so $a^* = b a(t) + \lambda_1 v(t) - \lambda_2 g$

so the control policy will be

$$a^* = \begin{cases} 0 & \text{if } b \leq 0 \\ 1 & \text{if } b > 0 \end{cases}$$

By logic we can state that the acceleration of the lander is active during the ending phase of the landing, so

$a^* = 0$ from $t \in [0, t_1]$

and $a^* = 1$ from $t \in (t_1, t^*]$

when ~~t_1 is~~ t_1 is the time when the acceleration changes or thrust is active, and t^* is the ~~term~~ terminal time of the landing.

so from $t \in [0, t_1]$, $b \leq 0$ and

from $t \in (t_1, t^*]$, $b > 0$.

so ~~the~~ in b is increasing with ~~time~~ time, which may mean that b is monotonous, we can prove this

$$b = -1 + \lambda_2 / m(t) - \lambda_3 k$$

$$\dot{b} = \frac{db}{dt} = \frac{\dot{\lambda}_2}{m(t)} - \frac{\lambda_2}{m(t)^2}(\dot{m}(t)) - \dot{\lambda}_3 k$$

To get values of $\dot{\lambda} \Rightarrow \dot{\lambda} = -\partial H / \partial x$

$$\dot{\lambda} = \begin{bmatrix} \dot{\lambda}_1 \\ \dot{\lambda}_2 \\ \dot{\lambda}_3 \end{bmatrix} = \begin{bmatrix} -\partial H / \partial h \\ -\partial H / \partial v \\ -\partial H / \partial m \end{bmatrix} = \begin{bmatrix} 0 \\ -\lambda_1 \\ \lambda_2 a / m^2 \end{bmatrix}$$

so, $\dot{\lambda}_2 = -\lambda_1$ & $\dot{\lambda}_3 = \dfrac{\lambda_2 a}{m(t)^2}$

We can substitute this in ~~db/dt~~ $\dot{b}$, as

$$\dot{b} = \frac{-\lambda_1}{m(t)} - \frac{\lambda_2}{m(t)^2} \cdot \dot{m}(t) - \frac{\lambda_2 a k}{m^2(t)}$$

from the dynamics, we know $\dot{m}(t) = -a(t) \cdot k$

$$\text{so } \dot{b} = \frac{-\lambda_1}{m(t)} - \cancel{\frac{\lambda_2 (-k a(t))}{m(t)^2}} - \cancel{\frac{\lambda_2 a(t) k}{m(t)^2}}$$

$$\Rightarrow \dot{b} = \frac{-\lambda_1}{m(t)}$$

As, $m(t)$ is always +ve, and we have equality constraints in the dynamics, the lagrangian multipliers can be negative.

so $\dot{b} = -\dfrac{\lambda_1}{m(t)}$ is +ve is $m(t)$ is +ve & λ_1 is -ve

This supports our solution as this proves $\dot{b}$ is +ve and ~~in~~ so b is always increasing and is monotonous in nature.