```
clc;
close all;

%%
L=1000;
n=10000;
theta_s=10; %It costs the reserve (theta_s/2)*x^2 to produce
x amount of power.
maxPrice=50;

%%
proposedMechanismCost=Inf(maxPrice,1);
postedPriceCost=Inf(maxPrice,1);

for price=1:maxPrice
    price

    opt_social_cost=Inf(L,1);
    postedPriceSocialCost=Inf(L,1);

    for dayIndex=1:L

        delta=randi(10,n,1); %n random parameters uniformly
distributed in {1,...,20). It costs Load i
(delta_i/2)*x_i^{2} to shed x_i power.
        DR_req=n*10*rand(1); %DR required. Each load should
shed about 10*rand(1) Watts.

        %%

opt_social_cost(dayIndex)=proposedMechanism(delta,theta_s,DR_
req);

postedPriceSocialCost(dayIndex)=postedPriceMechanism(price,de
lta,theta_s,DR_req);

    end

    proposedMechanismCost(price)=mean(opt_social_cost);
    postedPriceCost(price)=mean(postedPriceSocialCost);

end

figure(1);
```

```
plot(1:maxPrice,postedPriceCost,'-.k',1:maxPrice,proposedMech
anismCost,'-k','linewidth',1.5);
grid;
xlabel('Price');
ylabel('Social Cost');
legend('Posted Price Mechanism','Proposed Mechanism');

function SC_opt=proposedMechanism(delta,theta_s,DR_req)
n=size(delta,2);
H=zeros(n+1,n+1);
H(n+1,n+1)=theta_s;
for i=1:n
    H(i,i)=delta(i);
end
f=zeros(n+1,1);
Aeq=ones(1,n+1);
beq=DR_req;
Aineq=zeros(1,n+1);
bineq=0;
lb=-Inf(n+1,1);
ub=Inf(n+1,1);
options =  optimset('Display','off');
[~,SC_opt]=quadprog(H,f,Aineq,bineq,Aeq,beq,lb,ub,[],options)
;
end

function
SC_price=postedPriceMechanism(price,delta,theta_s,DR_req)

n=size(delta,2);
x_opt=zeros(n,1);
for i=1:n
    x_opt(i)=price/delta(i);
end
tot_shed=sum(x_opt);
gs=(DR_req)-(tot_shed);
cs=(theta_s/2)*((gs)^2);
SC_price=cs;
for i=1:n
    SC_price=(SC_price)+((delta(i)/2)*((x_opt(i))^2));
end
end
```