```
%------------IDA Simulation--------------%
clear
clc
%------------Parameters setting--------------%
M=5;            %Number of RSUs%
N=4;            %Number of EVs%
epsilon1=0.001; %Convergence Coefficient%
epsilon2=0.001; %Convergence Coefficient%
s=0.05;         %Step Size%
flag=0;
t=1

for m=1:1:M
    for n=1:1:N
        mu(m,n,t)=rand;
        lambda(n,t)=rand;
    end
end

while flag == 0
    judge=zeros(M,N);
    t=t+1;
    bid1=[0:0.0001:10];
    for m=1:1:M
        for n=1:1:N
            u1=log(1+(bid1/mu(m,n,t-1)))-bid1;
            [utility1(m,n),optbid1(m,n)]=max(u1);
            u(m,n,t)=0+(optbid1(m,n).*0.0001);
        end;
    end
    bid2=[0:0.0001:10];
    for m=1:1:M
        for n=1:1:N
            u2=(((mu(m,n,t-1)-lambda(n,t-1))^2)./bid2)-
(exp(((mu(m,n,t-1)-lambda(n,t-1))./bid2))-1);
            [utility2(n,m),optbid2(n,m)]=max(u2) ;
            u4(n,m,t)=utility2(n,m);
            v(n,m,t)=0+(optbid2(n,m)*0.0001);
        end
    end
    for m=1:1:M
        for n=1:1:N
            x(m,n,t)=u(m,n)./mu(m,n,t-1);
            y(n,m,t)=(mu(m,n,t-1)-lambda(n,t-1))./v(n,m,t-1);
        end
    end
    temp=sum(y,2);
    for m=1:1:M
        for n=1:1:N
            lambda(n,t)=max(lambda(n,t-1)+s*(temp(n)-1),0);
            mu(m,n,t)=max(mu(m,n,t-1)+s*(x(m,n,t)-y(n,m,t)),0);
        end
```

```
    end
%------------Convergence Results-------------%
    for m=1:1:M
        for n=1:1:N
            if abs((u(m,n,t)-u(m,n,t-1)))<epsilon1 && abs(v(n,m,t)-
v(n,m,t-1))<epsilon2
                judge(m,n)=1;
            end
            if isequal(judge,ones(M,N))
                flag=0;
            end
       end
    end
end
```

Published with MATLAB® R2021a

```
%------------PAM-N Simulation-------------%
clear
clc
%------------Parameters setting-------------%
M=5;            %Number of RSUs%
N=4;            %Number of EVs%
epsilon1=0.001; %Convergence Coefficient%
epsilon2=0.001; %Convergence Coefficient%
s=0.05;         %Step Size%
flag=0;
t=1

for m=1:1:M
    for n=1:1:N
        mu(m,n,t)=rand;
        lambda(n,t)=rand;
    end
end

while flag == 0
    judge=zeros(M,N);
    t=t+1;
    bid1=[0:0.0001:10];
    for m=1:1:M
        for n=1:1:N
            u1=log(1+(bid1/mu(m,n,t-1)))- bid1;
            [utility1(m,n),optbid1(m,n)]=max(u1);
            u(m,n,t)=0+(optbid1(m,n).*0.0001);
        end;
    end
    bid2=[0:0.0001:10];
    for m=1:1:M
        for n=1:1:N
            u2=(((mu(m,n,t-1)-lambda(n,t-1))^2)./bid2)-
 (exp(((mu(m,n,t-1)-lambda(n,t-1))./bid2))-1);
            [utility2(n,m),optbid2(n,m)]=max(u2) ;
            u4(n,m,t)=utility2(n,m);
            v(n,m,t)=0+(optbid2(n,m)*0.0001);
        end
    end
    temp_u=sum(u,2)
    temp_v=sum(v,2)
    C=sqrt(sum(temp_u).*sum(temp_v))
    for m=1:1:M
        for n=1:1:N
            middle_uv=u(m,n,t)*v(n,m,t);
            if middle_uv<C
            x(m,n,t) = 0
            else
            x(m,n,t) =2*u(m,n,t)/(t+sqrt(t^2+4*(u(m,n,t)/v(n,m,t))));
            end
```

```
            y(n,m,t) = x(m,n,t) + sqrt( x(m,n,t) + 4*(u(m,n,t)/
v(n,m,t)));
        end
    end
    temp=sum(y,2);
    for m=1:1:M
        for n=1:1:N
            lambda(n,t)=max(lambda(n,t-1)+s*(temp(n)-1),0);
            mu(m,n,t)=max(mu(m,n,t-1)+s*(x(m,n,t)-y(n,m,t)),0);
        end
    end
%------------Convergence Results--------------%
    for m=1:1:M
        for n=1:1:N
            if abs((u(m,n,t)-u(m,n,t-1)))<epsilon1 && abs(v(n,m,t)-
v(n,m,t-1))<epsilon2
                judge(m,n)=1;
            end
            if isequal(judge,ones(M,N))
                flag=0;
            end
       end
    end
end
```

Published with MATLAB® R2021a

```
%------------PAM-S Simulation-------------%
clear
clc
%------------Parameters setting-------------%
M=5;            %Number of RSUs%
N=4;            %Number of EVs%
epsilon1=0.001; %Convergence Coefficient%
epsilon2=0.001; %Convergence Coefficient%
s=0.05;         %Step Size%
flag=0;
t=1

for m=1:1:M
    for n=1:1:N
        mu(m,n,t)=rand;
        lambda(n,t)=rand;
    end
end

while flag == 0
    judge=zeros(M,N);
    t=t+1;
    bid2=[0:0.0001:10];
    for m=1:1:M
        for n=1:1:N
            u2=(((mu(m,n,t-1)-lambda(n,t-1))^2)./bid2)-
(exp(((mu(m,n,t-1)-lambda(n,t-1))./bid2))-1);
            [utility2(n,m),optbid2(n,m)]=max(u2) ;
            u4(n,m,t)=utility2(n,m);
            v(n,m,t)=0+(optbid2(n,m)*0.0001);
        end
    end
    for m=1:1:M
        for n=1:1:N
            if v(n,m,t)>0
                u(m,n,t)=((v(n,m,t)/2)^2)/(v(n,m,t))
            else
                u(m,n,t)=0
            end
        end
    end
    for m=1:1:M
        for n=1:1:N
            x(m,n,t)=u(m,n)./mu(m,n,t-1);
            y(n,m,t)=(mu(m,n,t-1)-lambda(n,t-1))./v(n,m,t);
        end
    end
    temp=sum(y,2);
    for m=1:1:M
        for n=1:1:N
            lambda(n,t)=max(lambda(n,t-1)+s*(temp(n)-1),0);
            mu(m,n,t)=max(mu(m,n,t-1)+s*(x(m,n,t)-y(n,m,t)),0);
```

```
        end
    end
%------------Convergence Results-------------%
    for m=1:1:M
        for n=1:1:N
            if abs((u(m,n,t)-u(m,n,t-1)))<epsilon1 && abs(v(n,m,t)-
v(n,m,t-1))<epsilon2
                judge(m,n)=1;
            end
            if isequal(judge,ones(M,N))
                flag=0;
            end
       end
    end
end
```

Published with MATLAB® R2021a

```
%------------Price-first Simulation-------------%
clear
clc
%------------Parameters setting-------------%
M=5;            %Number of RSUs%
N=4;            %Number of EVs%
epsilon1=0.001; %Convergence Coefficient%
epsilon2=0.001; %Convergence Coefficient%
s=0.05;         %Step Size%
flag=0;
t=1

for m=1:1:M
    for n=1:1:N
        mu(m,n,t)=rand;
        lambda(n,t)=rand;
    end
end

while flag == 0
    judge=zeros(M,N);
    t=t+1;
    bid1=[0:0.0001:10];
    for m=1:1:M
        for n=1:1:N
            u1=log(1+(bid1/mu(m,n,t-1)))-bid1;
            [utility1(m,n),optbid1(m,n)]=max(u1);
            u(m,n,t)=0+(optbid1(m,n).*0.0001);
        end;
    end
    bid2=[0:0.0001:10];
    for m=1:1:M
        for n=1:1:N
            u2=(((mu(m,n,t-1)-lambda(n,t-1))^2)./bid2)-
(exp(((mu(m,n,t-1)-lambda(n,t-1))./bid2))-1);
            [utility2(n,m),optbid2(n,m)]=max(u2) ;
            u4(n,m,t)=utility2(n,m);
            v(n,m,t)=0+(optbid2(n,m)*0.0001);
        end
    end
    for m=1:1:M
        for n=1:1:N
            x(m,n,t)=u(m,n)./mu(m,n,t-1);
            y(n,m,t)=u(m,n);
        end
    end
    temp=sum(y,2);
    for m=1:1:M
        for n=1:1:N
            lambda(n,t)=max(lambda(n,t-1)+s*(temp(n)-1),0);
            mu(m,n,t)=max(mu(m,n,t-1)+s*(x(m,n,t)-y(n,m,t)),0);
        end
```

```
    end
%------------Convergence Results-------------%
    for m=1:1:M
        for n=1:1:N
            if abs((u(m,n,t)-u(m,n,t-1)))<epsilon1 && abs(v(n,m,t)-
v(n,m,t-1))<epsilon2
                judge(m,n)=1;
            end
            if isequal(judge,ones(M,N))
                flag=0;
            end
       end
    end
end
```

Published with MATLAB® R2021a

```
%------------On-demand Simulation-------------%
clear
clc
%------------Parameters setting-------------%
M=5;            %Number of RSUs%
N=4;            %Number of EVs%
epsilon1=0.001; %Convergence Coefficient%
epsilon2=0.001; %Convergence Coefficient%
s=0.05;         %Step Size%
flag=0;
t=1

for m=1:1:M
    for n=1:1:N
        mu(m,n,t)=rand;
        lambda(n,t)=rand;
    end
end

while flag == 0
    judge=zeros(M,N);
    t=t+1;
    bid1=[0:0.0001:10];
    for m=1:1:M
        for n=1:1:N
            u1=log(1+(bid1/mu(m,n,t-1)))-bid1;
            [utility1(m,n),optbid1(m,n)]=max(u1);
            u(m,n,t)=0+(optbid1(m,n).*0.0001);
        end;
    end
    bid2=[0:0.0001:10];
    for m=1:1:M
        for n=1:1:N
            u2=(((mu(m,n,t-1)-lambda(n,t-1))^2)./bid2)-
(exp(((mu(m,n,t-1)-lambda(n,t-1))./bid2))-1);
            [utility2(n,m),optbid2(n,m)]=max(u2) ;
            u4(n,m,t)=utility2(n,m);
            v(n,m,t)=0+(optbid2(n,m)*0.0001);
        end
    end
    for m=1:1:M
        for n=1:1:N
            x(m,n,t)=u(m,n)./mu(m,n,t-1);
            y(n,m,t)=u(m,n)./mu(m,n,t-1);
        end
    end
    temp=sum(y,2);
    for m=1:1:M
        for n=1:1:N
            lambda(n,t)=max(lambda(n,t-1)+s*(temp(n)-1),0);
            mu(m,n,t)=max(mu(m,n,t-1)+s*(x(m,n,t)-y(n,m,t)),0);
        end
```

```
    end
%------------Convergence Results-------------%
    for m=1:1:M
        for n=1:1:N
            if abs((u(m,n,t)-u(m,n,t-1)))<epsilon1 && abs(v(n,m,t)-
v(n,m,t-1))<epsilon2
                judge(m,n)=1;
            end
            if isequal(judge,ones(M,N))
                flag=0;
            end
       end
    end
end
```

Published with MATLAB® R2021a