N=10000;m0=3;m=3;  
adjacent\_matrix=sparse(m0,m0);  
for i=1:m0  
for j=1:m0  
if j==i  
adjacent\_matrix(i,j)=1;  
end  
end  
end  
adjacent\_matrix=sparse(adjacent\_matrix);  
node\_degree=zeros(1,m0+1);  
node\_degree(2:m0+1)=sum(adjacent\_matrix);  
for iter=4:N  
iter  
total\_degree=2\*m\*(iter-4)+6;  
cum\_degree=cumsum(node\_degree);  
choose=zeros(1,m)  
%%%  
r1=rand(1)\*total\_degree;  
for i=1:iter-1  
if(r1>=cum\_degree(i))&(r1<cum\_degree(i+1))  
choose(1)=i;  
break  
end   
end  
choose(1)%%  
r2=rand(1)\*total\_degree;  
for i=1:iter-1  
if(r2>=cum\_degree(i))&(r2<cum\_degree(i+1))  
choose(2)=i;  
break  
end  
end  
while choose(2)==choose(1)  
r2=rand(1)\*total\_degree;  
for i=1:iter-1  
if(r2>=cum\_degree(i))&(r2<cum\_degree(i+1))  
choose(2)=i;  
break  
end  
end  
end  
choose(2) %%%  
r3=rand(1)\*total\_degree;  
for i=1:iter-1  
if(r3>=cum\_degree(i))&(r3<cum\_degree(i+1))  
choose(3)=i;  
break  
end  
end  
while(choose(3)==choose(1))|(choose(3)==choose(2))  
r3=rand(1)\*total\_degree;  
for i=1:iter-1  
if(r3>=cum\_degree(i))&(r3<cum\_degree(i+1))  
choose(3)=i;  
break  
end  
end  
end  
choose(3)%%%  
for k=1:m  
z=choose(k)  
adjacent\_matrix(z,iter)=1;  
adjacent\_matrix(iter,z)=1;  
end  
node\_degree=zeros(1,iter+1);  
node\_degree(2:iter+1)=sum(adjacent\_matrix);  
end  
save data adjacent\_matrix;