

parPosGen (generic function with 1 method)

```
1 #####generate particle position#####
2 function
  parPosGen(center::Bool,pNo::Integer,foldNo::Integer,ringNo::Integer,Rc::Array{Float64
  })
3 parPosTmp=Matrix{Float64}(undef,pNo,3);
4 rcTmp=deepcopy(Rc);
5 if center
6     parPosTmp[1,:]=[0.0 0 0];
7     for i in 1:ringNo
8         for j in 1:foldNo
9             parPosTmp[j+(i-1)*foldNo+1,1]=rcTmp[i]*cos((j-1)*2π/foldNo);
10            parPosTmp[j+(i-1)*foldNo+1,2]=rcTmp[i]*sin((j-1)*2π/foldNo);
11            parPosTmp[j+(i-1)*foldNo+1,3]=0.0;
12        end
13    end
14 else
15     for i in 1:ringNo
16         for j in 1:foldNo
17             parPosTmp[j+(i-1)*foldNo,1]=rcTmp[i]*cos((j-1)*2π/foldNo);
18             parPosTmp[j+(i-1)*foldNo,2]=rcTmp[i]*sin((j-1)*2π/foldNo);
19             parPosTmp[j+(i-1)*foldNo,3]=0.0;
20         end
21     end
22 end
23 return parPosTmp
24 end
```

mdS (generic function with 1 method)

```

1 #####molecule dynamics for equilibrium with rotational symmetries#####
2 function
  mdS(center::Bool,pNo::Integer,foldNo::Integer,ringNo::Integer,Rc::Array{Float64},Rs::
    Float64)
3 precSet=10^-6; #set the precision of equilibrium position
4 rndDigNo=9; #digit No. of round force.
5 minStepTmp=0.0002; #minimal step
6 maxStepTmp=minStepTmp*10; #maximal step
7 maxLopNo=500; #max loop No to find equilibrium position
8 dimension=3; #for 3D
9 pp=29.0; #particle density
10 parPosTmp=Matrix{Float64}(undef,pNo,3);
11 parPosData=Array{Float64}(undef,maxLopNo*10,dimension,pNo);
12 rcTmp=deepcopy(Rc);
13 parPosTmp=parPosGen(center,pNo,foldNo,ringNo,rcTmp);
14 forceTmp=round.(forcePackLow(Rs,parPosTmp);digits=rndDigNo);
15 disTmp2=round.(movDis(forceTmp,Rs,pp);digits=rndDigNo);
16 if findmax(abs,disTmp2)[1]==0
17     println("particles are in EQUILIBRIUM positions!")
18     return parPosTmp
19 end
20 countTmp=1;
21 countTmp2=1;
22 while minStepTmp>=precSet
23     minStepTmp=minStepTmp/5;
24     maxStepTmp=maxStepTmp/5;
25     for lopTmp in 1:maxLopNo
26         disTmp=round.(movDis(forceTmp,Rs,pp);digits=rndDigNo);
27         if findmax(abs,disTmp)[1]==0
28             println("particles are in EQUILIBRIUM positions!")
29             return parPosTmp
30         end
31         if ensMov(disTmp)==1
32             println("EQUILIBRIUM, ENSEMBLE MOVEMENT")
33             println("Rs: "*string(Rs))
34             println(parPosTmp)
35             return parPosTmp
36         end
37         disTmp=disUpCheck(disTmp,maxStepTmp);
38         disTmp=disLowCheck(disTmp,minStepTmp);
39         disTmp=reScal(disTmp);
40         if findmax(disTmp.*disTmp2)[1]<=0 #condition of equilibrium
41             println("particles are in EQUILIBRIUM positions!")
42             println("precision:",maxStepTmp)
43             for i in 1:ringNo
44                 if center
45                     rcTmp[i]=rcTmp[i]+disTmp[(1+(i-1)*foldNo)*3+1]/2;
46                 else
47                     rcTmp[i]=rcTmp[i]+disTmp[(i-1)*foldNo*3+1]/2;
48                 end
49             end
50             parPosTmp=round.
              (parPosGen(center,pNo,foldNo,ringNo,rcTmp);digits=rndDigNo);

```

```

51         disTmp2=deepcopy(disTmp);
52         break
53     else
54         for i in 1:ringNo
55             if center
56                 rcTmp[i]=rcTmp[i]+disTmp[(1+(i-1)*foldNo)*3+1,1];
57             else
58                 rcTmp[i]=rcTmp[i]+disTmp[(i-1)*foldNo*3+1,1];
59             end
60         end
61         parPosTmp=round.
        (parPosGen(center,pNo,foldNo,ringNo,rcTmp);digits=rndDigNo);
62         for iTmp in 1:pNo
63             for jTmp in 1:dimension
64                 parPosData[countTmp,jTmp,iTmp]=parPosTmp[iTmp,jTmp];
65             end
66         end
67         disTmp2=deepcopy(disTmp);
68         parPosTmp=sepCheck(parPosTmp,Rs);
69         forceTmp=round.(forcePackLow(Rs,parPosTmp);digits=rndDigNo);
70     end
71     parPosData2=parPosData[1:countTmp,:,:];
72     plot3d(plottitles=countTmp);
73     println(["Max Displacement: " * string(maxStepTmp), "Total steps: " *
        string(countTmp),"RND No.: " * string(countTmp2),"Step No.: " *
        string(lopTmp)])
74     for iTmp in 1:pNo
75         display(path3d!
        (parPosData2[:,1,iTmp],parPosData2[:,2,iTmp],parPosData2[:,3,iTmp]))
76     end
77     println(disTmp)
78     println("current particle positions:")
79     println(parPosTmp)
80     countTmp+=1;
81     if lopTmp==maxLopNo
82         println("Loop No. reaches Maximum!")
83     end
84     end
85     countTmp2+=1;
86 end
87 return round.(parPosGen(center,pNo,foldNo,ringNo,rcTmp);digits=(rndDigNo-2))
88 end
89

```

