

0.1 Detailed Documentation of Simulation Framework

```
1 function p = activnet_gen(zet,L,mu,kap,lc,xi,ups,phi,psi,r,  
    sig,Dx,Dy,Df,Dw,ls,lf,tinc,tfin,nonlin)  
2 % generates an active network simulation and prints node  
    positions  
3 % at time steps. Parameters are defined as follows:  
4 %  
5 %   zet - medium viscosity  
6 %   L - length of the filament  
7 %   mu - compressional modulus of the filament  
8 %   kap - bending modulus of a filament if  $ls < L$   
9 %   lc - average distance between filament overlaps  
10 %   xi - frictional resistance between two overlapping  
    segments  
11 %   ups - motor force at filament overlaps  
12 %   phi - fraction of overlaps that receive a motor force  
13 %   psi - spatial variation in motor force (see below)  
14 %   sig - applied force  
15 %   Dx - x-dimension of domain  
16 %   Dy - y-dimension of domain  
17 %   Df -  
18 %   Dw - width of window in x-dimension where forces/  
    constraints are applied  
19 %   ls - length of filament segments  
20 %   lf - length of force falloff at end of filament (for  
    continuous forces)  
21 %   tinc - time increment to return solutions  
22 %   tfin - end time of simulation  
23 %   nonlin - nonlinear factor by which to make filament  
    stiffer by extension
```

```

24 %
25 %   See also SUM, PLUS.
26
27
28 %% this mess just ensures that any string input is converted
   to numbers
29     if(ischar(zet)); zet = str2num(zet); end;
30     if(ischar(L)); L = str2num(L); end;
31     if(ischar(mu)); mu = str2num(mu); end;
32     if(ischar(kap)); kap = str2num(kap); end;
33     if(ischar(lc)); lc = str2num(lc); end;
34     if(ischar(xi)); xi = str2num(xi); end;
35     if(ischar(ups)); ups = str2num(ups); end;
36     if(ischar(phi)); phi = str2num(phi); end;
37     if(ischar(psi)); psi = str2num(psi); end;
38     if(ischar(r)); r = str2num(r); end ;
39     if(ischar(sig)); sig = str2num(sig); end;
40     if(ischar(Dx)); Dx = str2num(Dx); end;
41     if(ischar(Dy)); Dy = str2num(Dy); end;
42     if(ischar(Df)); Df = str2num(Df); end;
43     if(ischar(Dw)); Dw = str2num(Dw); end;
44     if(ischar(ls)); ls = str2num(ls); end;
45     if(ischar(lf)); lf = str2num(lf); end;
46     if(ischar(tinc)); tinc = str2num(tinc); end;
47     if(ischar(tfin)); tfin = str2num(tfin); end;
48     if(ischar(nonlin)); nonlin = str2num(nonlin); end;
49     Dp = 1;
50     if(Df<0);Df=abs(Df);Dp = Df; end;
51 %     rng(abs(nonlin));
52
53 %% use inputs to calculate number of filaments to add

```

```

54     ncnt = ceil(L/lc)+1;
55     N = floor(2*Dx*Dy/lc/L);
56
57     nu=[];
58     if(ups>0)
59         nu = ups*double(rand(N,N)<phi).*(ones(N,N)-eye(N,N));
60         nu = (nu+nu')/2;
61     end
62
63     %% initialize network
64
65     p = zeros(N*ncnt,2);
66     for i=1:N
67         p((i-1)*ncnt+1,:) = [Dp*Dx*rand Dy*rand];
68         thet = rand*2*pi;
69         for j = 2:ncnt
70             p((i-1)*ncnt+j,:) = p((i-1)*ncnt+j-1,:)+L/(ncnt
71                 -1.0)*[cos(thet) sin(thet)];
72         end
73     end
74     if(nonlin<0)
75         p = zeros(N*ncnt,2);
76         for i=1:N
77             p((i-1)*ncnt+1,:) = [Dx*(0.2+0.6*rand) Dy
78                 *(0.2+0.6*rand)];
79             thet = rand*2*pi;
80             for j = 2:ncnt
81                 p((i-1)*ncnt+j,:) = p((i-1)*ncnt+j-1,:)+L/(
82                     ncnt-1.0)*[cos(thet) sin(thet)];
83             end
84         end
85     end

```

```

82         nonlin=-nonlin;
83     end
84
85     muN = nonlin;
86
87     p = [mod(p(:,1),Dx),mod(p(:,2),Dy)];
88
89     fileID = 1;
90     %% solve ode
91     z0 = reshape(p,1,[]);
92
93     if(tinc > 0.05/2/r)
94         tinc = 0.05/2/r;
95     end
96     tt = 0:tinc:tfin;
97
98     fprintf(fileID, '%.3f ',0);
99     for i=1:length(z0)
100         fprintf(fileID, ' %.4f ',z0(i));
101     end
102     fprintf(fileID, '\n');
103
104     activnet(N,tt,z0,zet,L,mu,muN,kap,xi,nu,psi,sig,Dx,Dy,Df,
105             Dw,Dp,ncnt,lf,r,tinc,fileID);
106
107 end

```