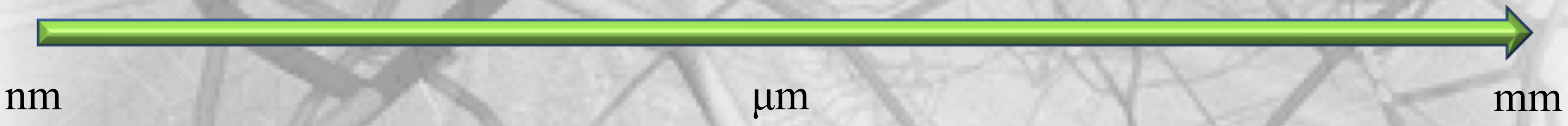
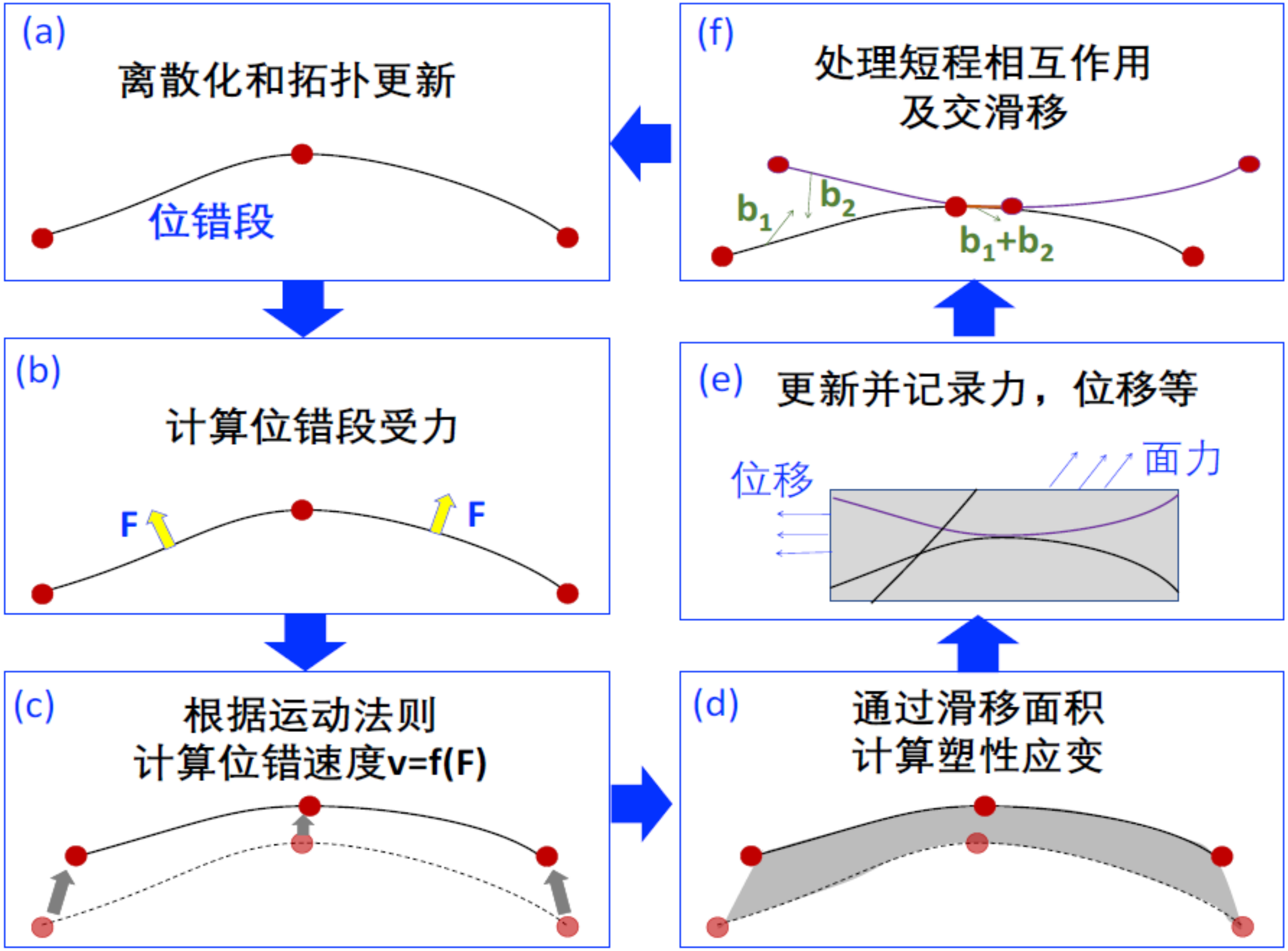
位错动力学



助教：罗士超

2020.1.13

位错动力学程序基本计算流程



(a) 离散化和拓扑更新

(b) 计算位错段受力

(c) 根据运动法则计算位错速度v=f(F)

(d) 通过滑移面积计算塑性应变

(e) 更新并记录力，位移等

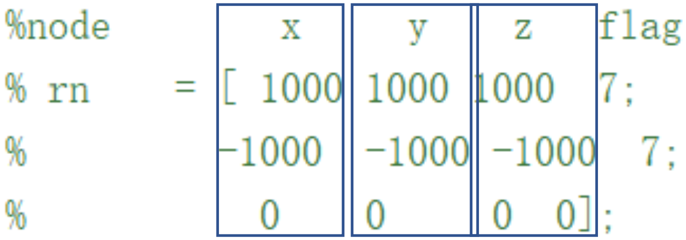
(f) 处理短程相互作用及交滑移

初始输入参数input

1.初始微结构的定义（以input_frank_read.m为例）

定义离散位错段上的节点：

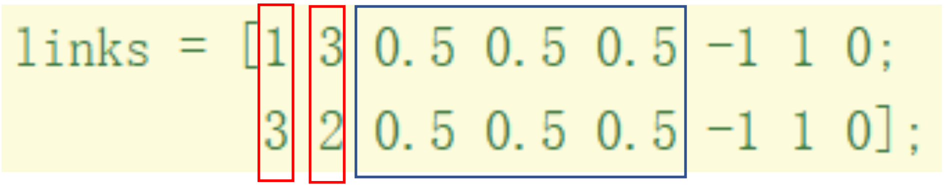
节点坐标



```
%node              x      y      z     flag
% rn        = [ 1000   1000   1000   7;
%              -1000  -1000  -1000    7;
%                  0      0      0   0];
```

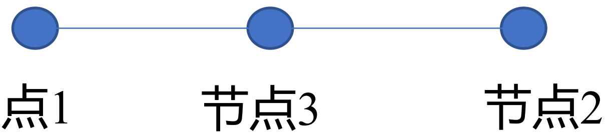
节点1	不可动
节点2	不可动
节点3	可动

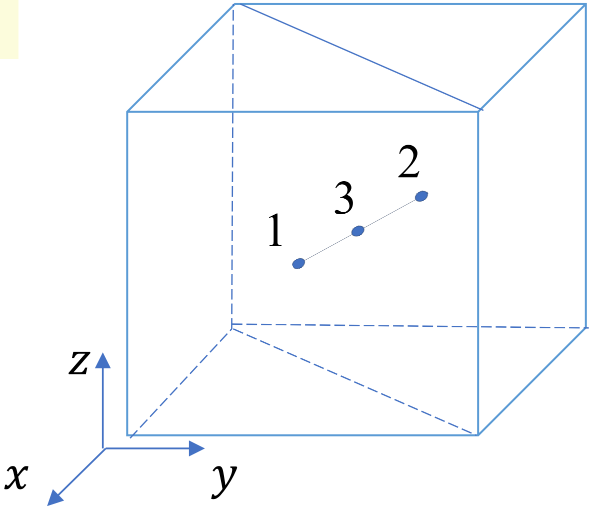
定义离散位错段： 终点 位错段所在滑移面



```
links = [1 3 0.5 0.5 0.5 -1 1 0;
         3 2 0.5 0.5 0.5 -1 1 0];
```

起点 位错的柏氏矢量





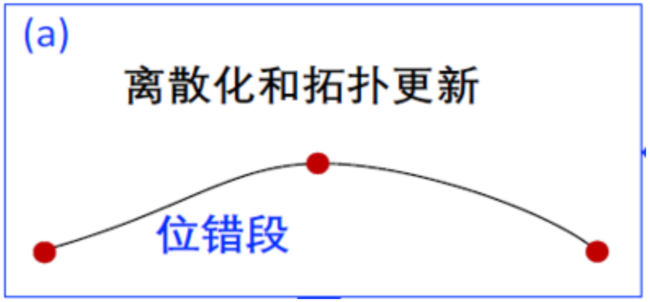
初始输入参数input

2.材料参数，拓扑等参数定义

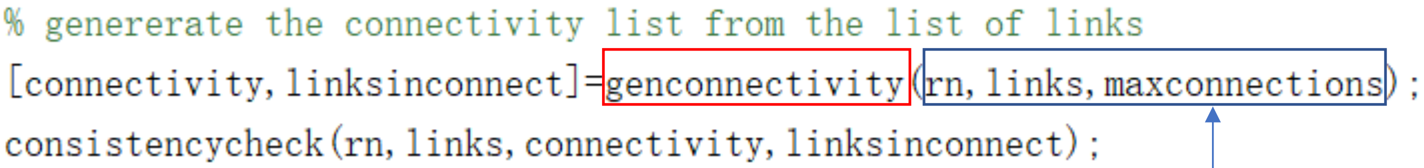
MU	剪切模量
NU	泊松比
maxconnections	
lmax	位错段的最大/最小长度
lmin	
areamin	
areamax	
a	位错核半径
Ec	单位长度的位错核能量
totalsteps	总时间步
dt0	最大步长
mobility	位错运动法则
integrator	数值方法

rann	湮没距离
rntol	精度
doremesh	拓扑 处理
docollision	
doseparation	
plotfreq	输出频率
plim	晶胞大小
appliedstress	外载
viewangle	
printfreq	输出频率
printnode	
rmax	一个时间步内节点所能运动的最大距离

位错的离散和拓扑

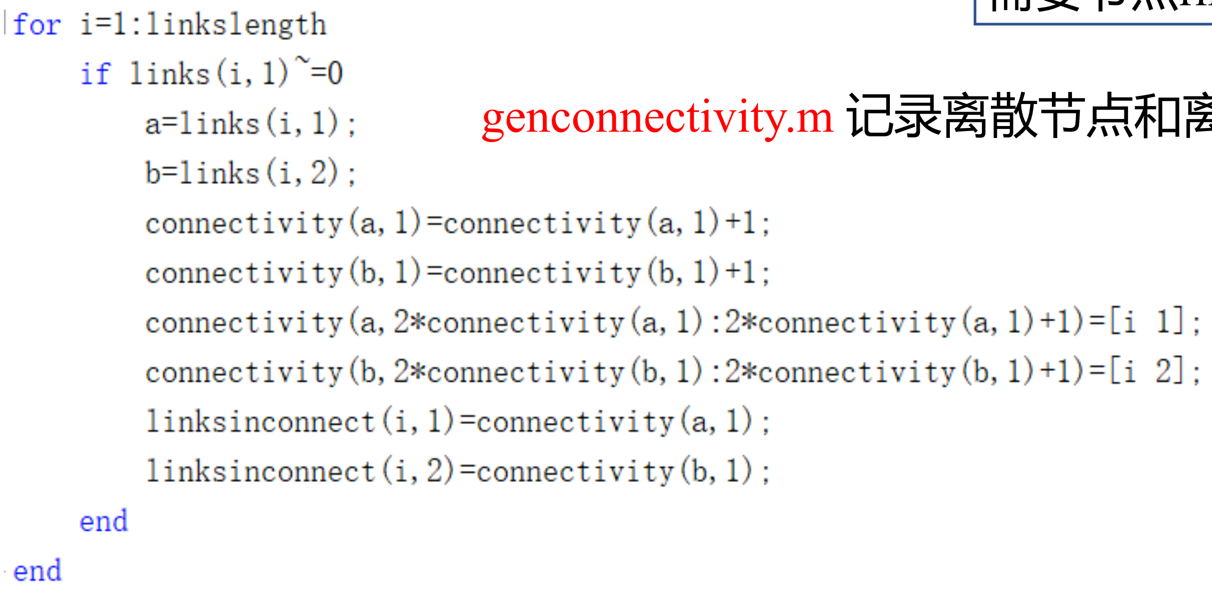


dd3d.m



```matlab
% genererate the connectivity list from the list of links
[connectivity,linksinconnect]=genconnectivity(rn,links,maxconnections);
consistencycheck(rn,links,connectivity,linksinconnect);
```

需要节点rn，段links信息的输入

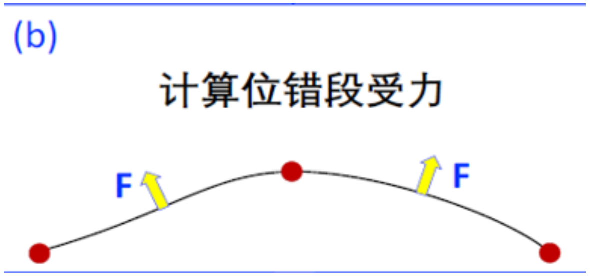


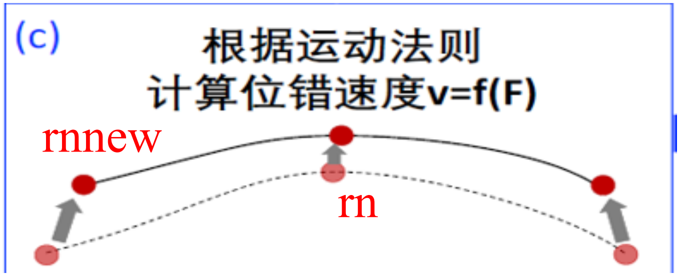
```matlab
for i=1:linkslength
    if links(i,1)~=0
        a=links(i,1);
        b=links(i,2);
        connectivity(a,1)=connectivity(a,1)+1;
        connectivity(b,1)=connectivity(b,1)+1;
        connectivity(a,2*connectivity(a,1):2*connectivity(a,1)+1)=[i 1];
        connectivity(b,2*connectivity(b,1):2*connectivity(b,1)+1)=[i 2];
        linksinconnect(i,1)=connectivity(a,1);
        linksinconnect(i,2)=connectivity(b,1);
    end
end
```

genconnectivity.m 记录离散节点和离散位错段的信息

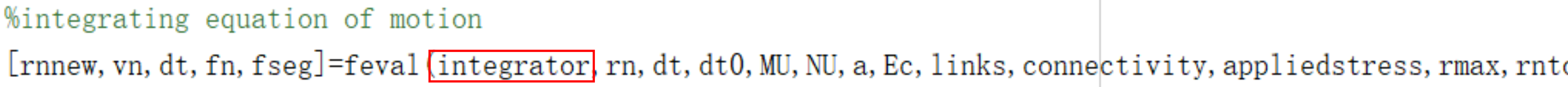
consistencycheck.m检查离散的位错段是否在物理和拓扑两方面上合理

计算节点速度vn，节点力fn，时间步dt，位错段的受fseg





dd3d.m

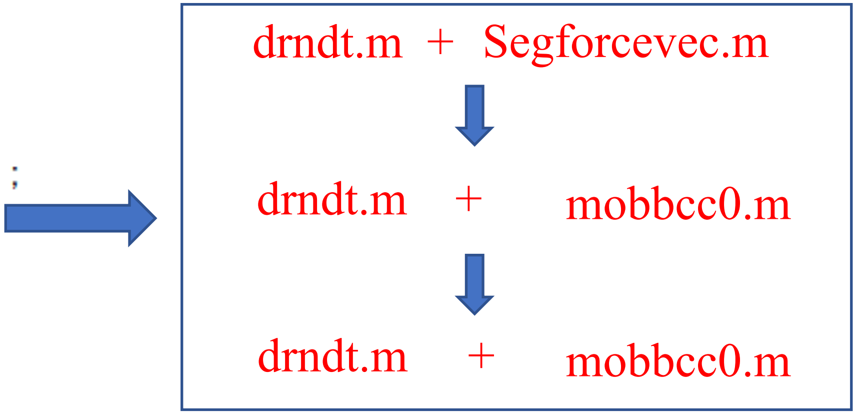


```
%integrating equation of motion
[rnnew, vn, dt, fn, fseg]=feval(integrator, rn, dt, dt0, MU, NU, a, Ec, links, connectivity, appliedstress, rmax, rntc
```

在Input中有定义

```
integrator=’int_trapezoid’;
```

int_trapezoid.m



drndt.m ＋ Segforcevec.m — 计算位错段速度

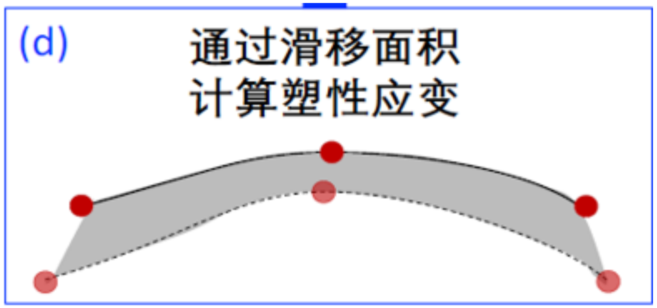
drndt.m ＋ mobbcc0.m — 计算节点力

drndt.m ＋ mobbcc0.m — 计算节点速度

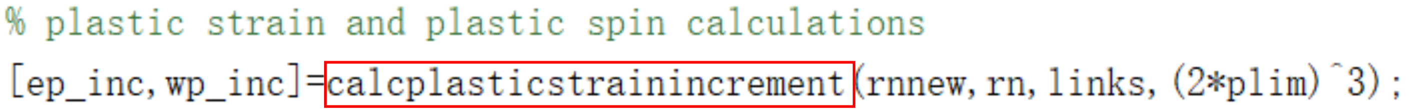
$$rnnew = rn + vn * dt$$

用向后的欧拉法计算$\vec{v} = \frac{d\vec{r}}{dt}$,为满足精度不断调整dt

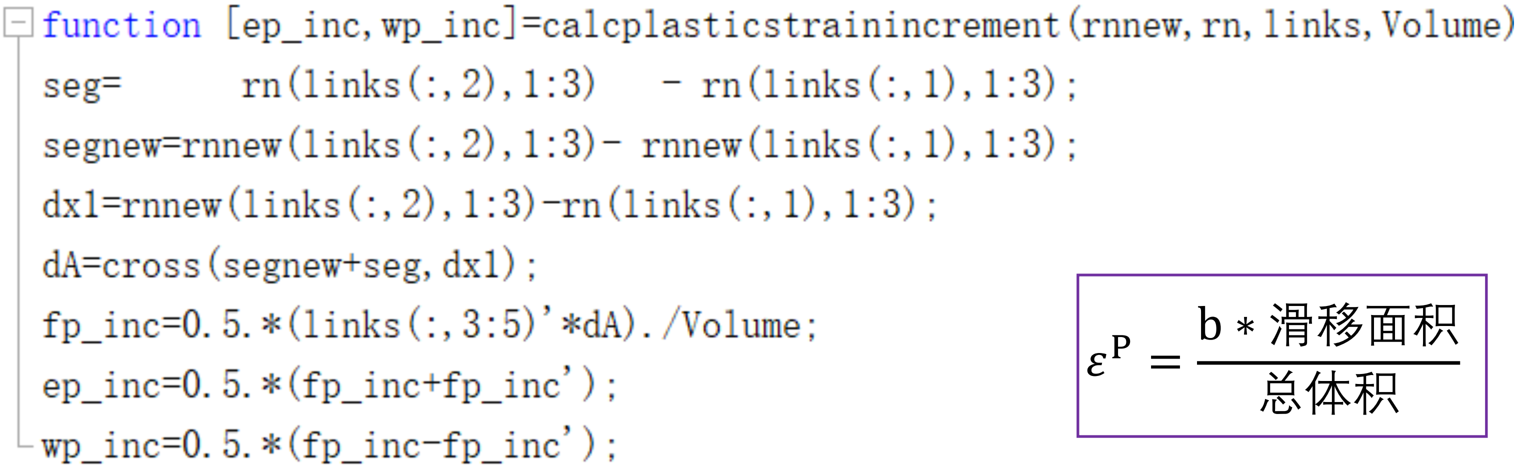
计算塑性变形



dd3d.m



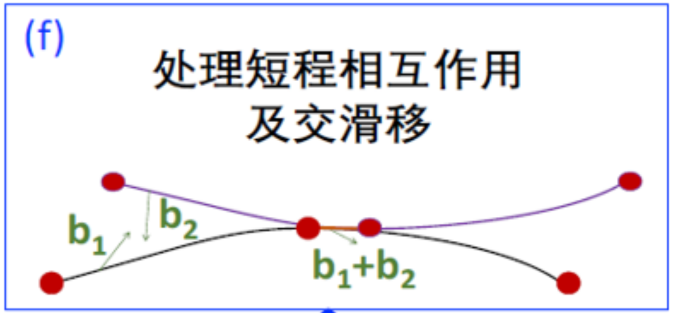
```
% plastic strain and plastic spin calculations
[ep_inc,wp_inc]=calcplasticstrainincrement(rnnew,rn,links,(2*plim)^3);
```



```
function [ep_inc,wp_inc]=calcplasticstrainincrement(rnnew,rn,links,Volume)
seg=        rn(links(:,2),1:3)    - rn(links(:,1),1:3);
segnew=rnnew(links(:,2),1:3)- rnnew(links(:,1),1:3);
dx1=rnnew(links(:,2),1:3)-rn(links(:,1),1:3);
dA=cross(segnew+seg,dx1);
fp_inc=0.5.*(links(:,3:5)'*dA)./Volume;
ep_inc=0.5.*(fp_inc+fp_inc');
wp_inc=0.5.*(fp_inc-fp_inc');
```

$$\boxed{\varepsilon^{P} = \frac{\text{b} * \text{滑移面积}}{\text{总体积}}}$$

位错几何拓扑处理



判断位错之前是否会反应

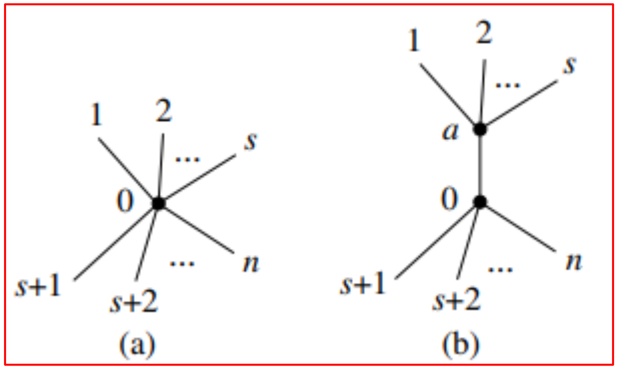
dd3d.m

```
if(doseparation)
    %spliting of nodes with 4 or more connections
    [rnnew,linksnew,connectivitynew,linksinconnectnew,fsegnew]=separation(rnnew,linksnew,connectivityr
```

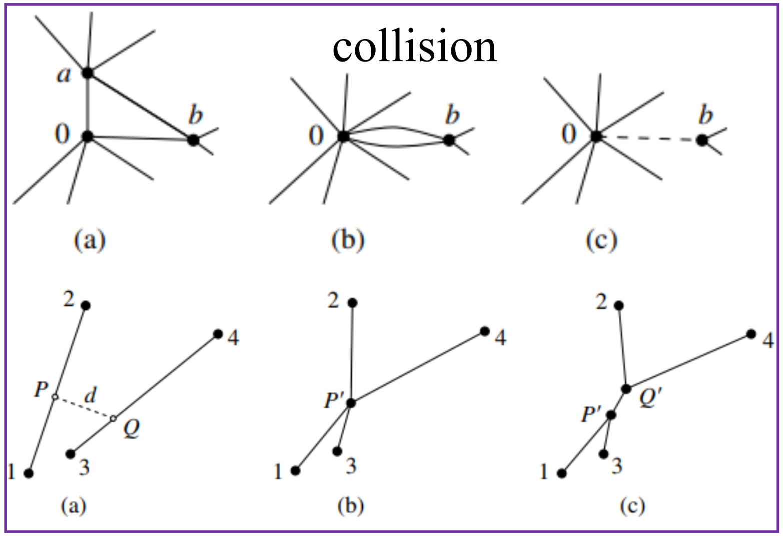
当一节点有大于3个分支位错段就会执行

```
end
if(docollision)
    %collision detection and handling
    [rnnew,linksnew,connectivitynew,linksinconnectnew,fsegnew]=collision(rnnew,linksnew,connectivityne
end
```

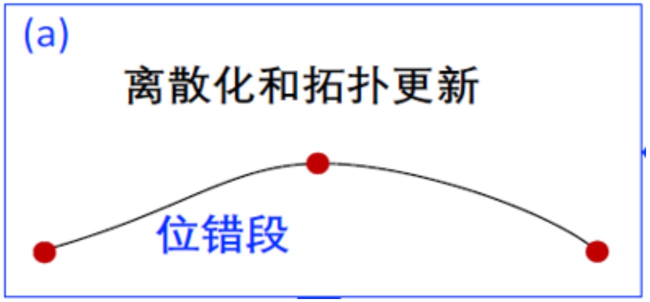
两点之间距离过近时执行

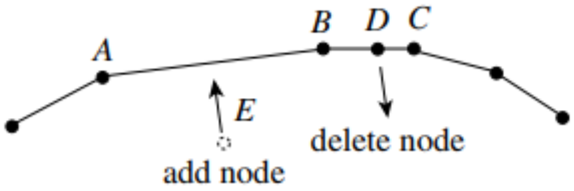


separation



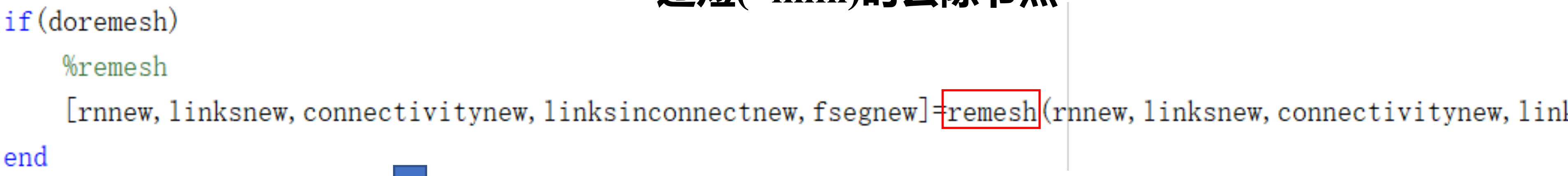
位错的离散和拓扑更新



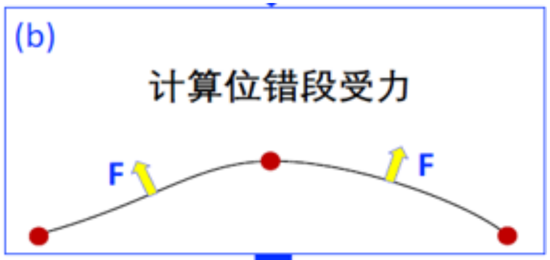


dd3d.m

拓扑更新：位错段过长（>lmax）时加入节点，过短(<lmin)时去除节点



```
if(doremesh)
    %remesh
    [rnnew, linksnew, connectivitynew, linksinconnectnew, fsegnew]=remesh(rnnew, linksnew, connectivitynew, lin
end
```



....

循环以上计算过程到最大输出步

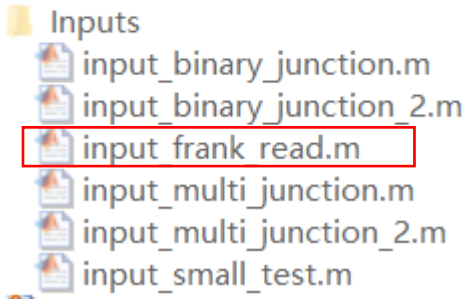
....

FR源位错动力学计算

运行程序很简单

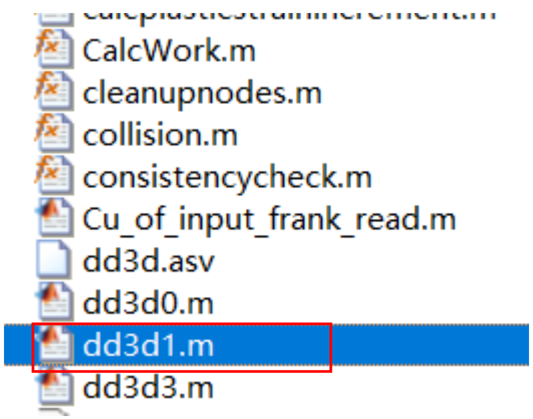
dd程序下载：http://micro.stanford.edu/~caiwei/Forum/2005-12-05-DDLab/

ddlab-2007-12-18.tar.gz



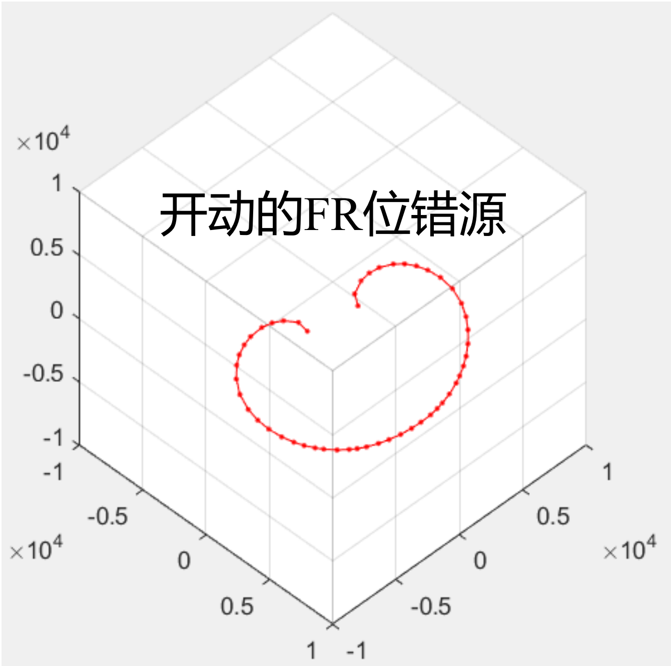
- Inputs
  - input_binary_junction.m
  - input_binary_junction_2.m
  - input_frank_read.m
  - input_multi_junction.m
  - input_multi_junction_2.m
  - input_small_test.m

在input文件夹下运行input_frank_read

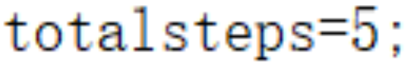


- calcplasticstrainincrement.m
- CalcWork.m
- cleanupnodes.m
- collision.m
- consistencycheck.m
- Cu_of_input_frank_read.m
- dd3d.asv
- dd3d0.m
- dd3d1.m
- dd3d3.m

在dd3d文件夹
下运行dd3d1



FR源位错动力学计算

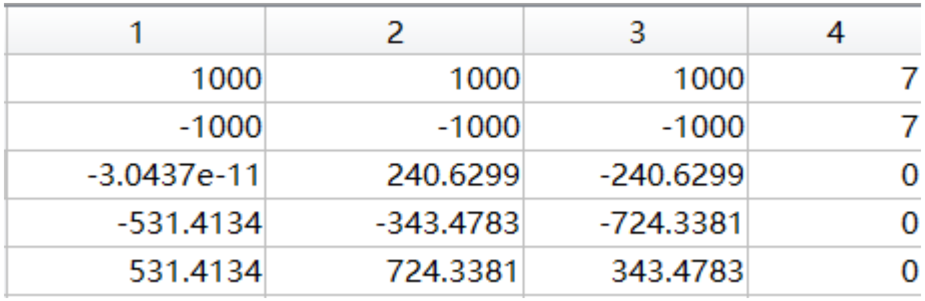


```
totalsteps=5;
```

先跑五步，简单介绍在程序中如何计算位错运动

五个节点

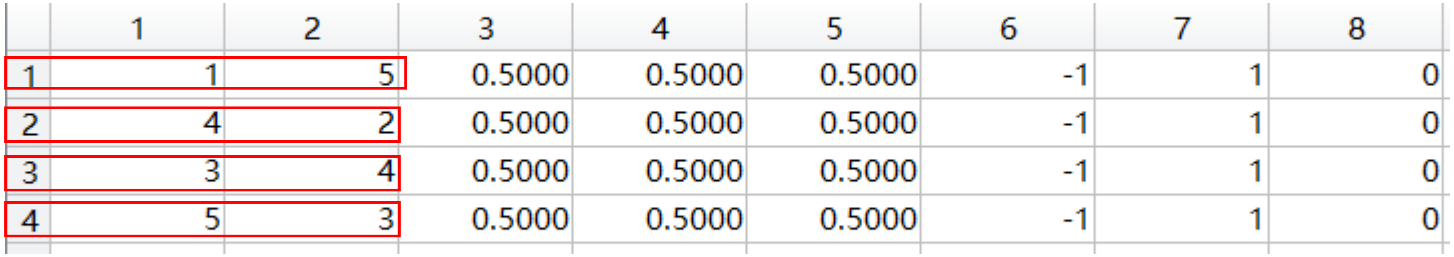
rn 5x4 double



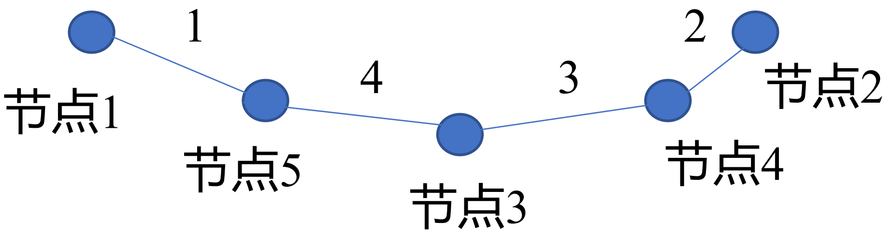
1	2	3	4
1000	1000	1000	7
-1000	-1000	-1000	7
-3.0437e-11	240.6299	-240.6299	0
-531.4134	-343.4783	-724.3381	0
531.4134	724.3381	343.4783	0

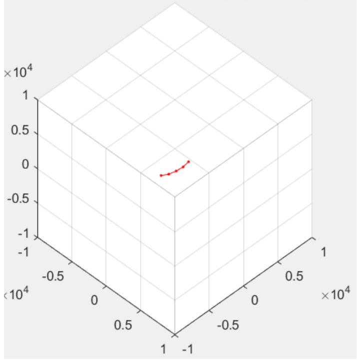
四条离散的位错段

links 4x8 double

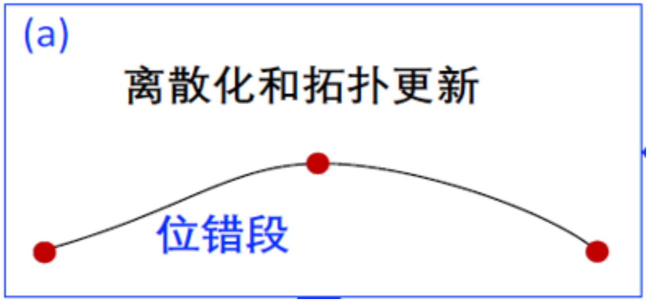


	1	2	3	4	5	6	7	8
1	1	5	0.5000	0.5000	0.5000	-1	1	0
2	4	2	0.5000	0.5000	0.5000	-1	1	0
3	3	4	0.5000	0.5000	0.5000	-1	1	0
4	5	3	0.5000	0.5000	0.5000	-1	1	0





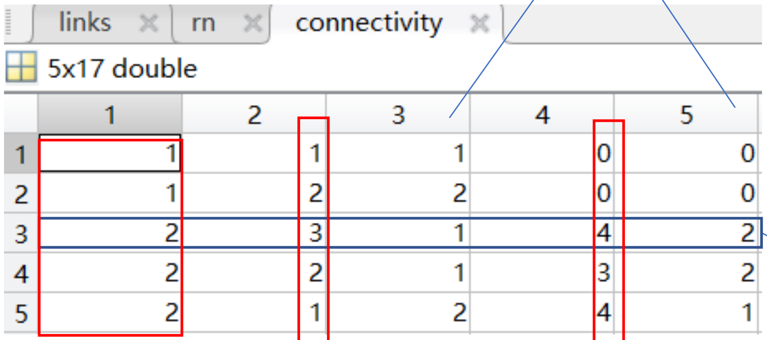
FR源位错动力学计算



dd3d.m

```
% genererate the connectivity list from the list of links
[connectivity,linksinconnect]=genconnectivity(rn,links,maxconnections);
```

是起点还是终点



5x17 double

	1	2	3	4	5
1	1	1	1	0	0
2	1	2	2	0	0
3	2	3	1	4	2
4	2	2	1	3	2
5	2	1	2	4	1

节点1
节点2
节点3
节点4
节点5

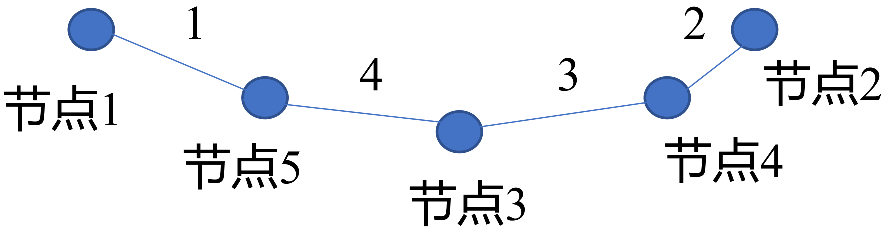
节点1在1号段上，是起点（1）

节点2在2号段上，是终点（2）

节点3在3号段上，是起点（1）
节点3在4号段上，是终点（2）

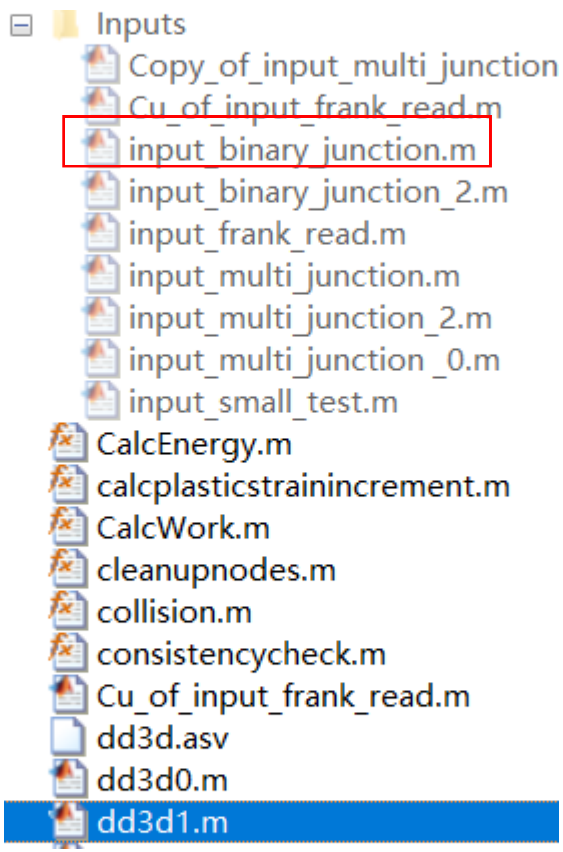
节点在几条
位错段上

节点在那条位错段上

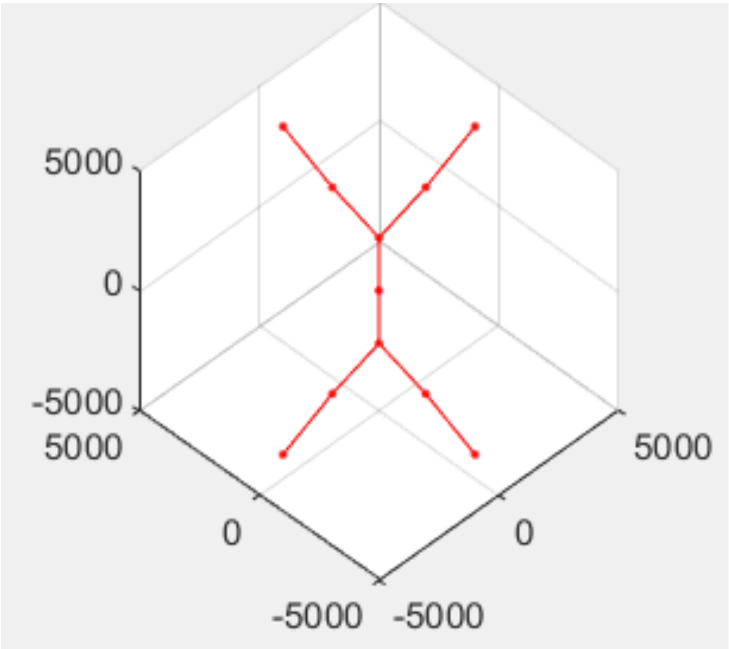


位错反应的例子

在input文件夹下运行input_binary_junction



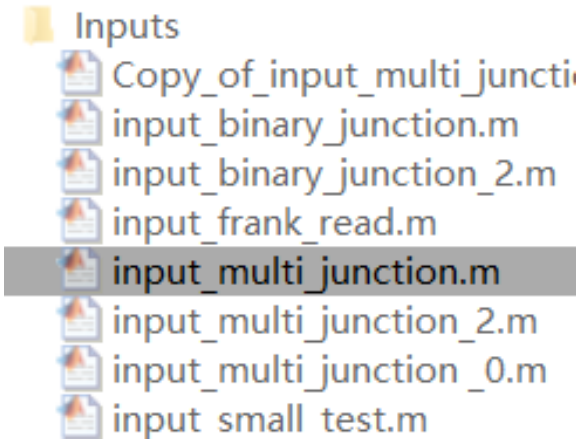
Inputs
Copy_of_input_multi_junction
Cu_of_input_frank_read.m
input_binary_junction.m
input_binary_junction_2.m
input_frank_read.m
input_multi_junction.m
input_multi_junction_2.m
input_multi_junction _0.m
input_small_test.m
CalcEnergy.m
calcplasticstrainincrement.m
CalcWork.m
cleanupnodes.m
collision.m
consistencycheck.m
Cu_of_input_frank_read.m
dd3d.asv
dd3d0.m
dd3d1.m



在dd3d_modified文件夹
下运行dd3d1

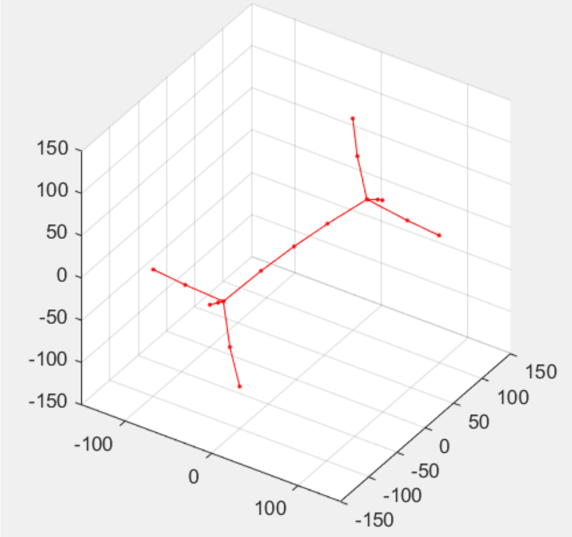
应力率加载的例子

在input文件夹下运行input_multi_junction

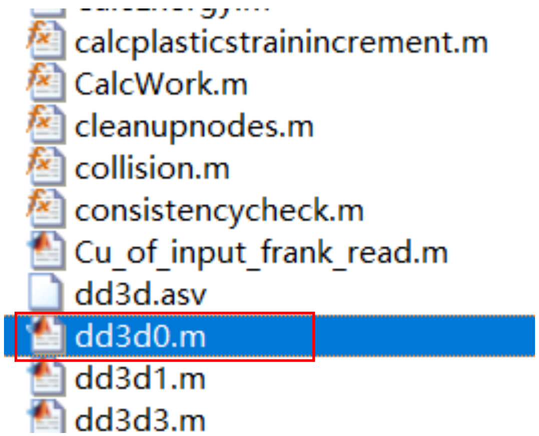


Inputs
Copy_of_input_multi_juncti
input_binary_junction.m
input_binary_junction_2.m
input_frank_read.m
input_multi_junction.m
input_multi_junction_2.m
input_multi_junction_0.m
input_small_test.m

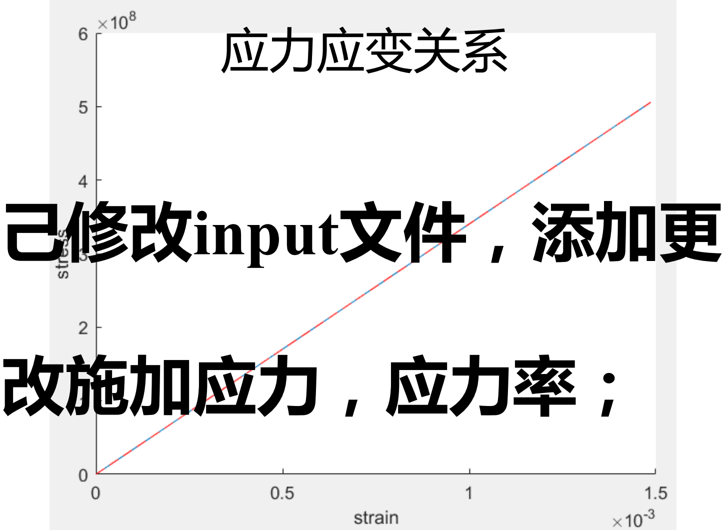
junction



在dd3d_modified文件夹下运行dd3d0



calcplasticstrainincrement.m
CalcWork.m
cleanupnodes.m
collision.m
consistencycheck.m
Cu_of_input_frank_read.m
dd3d.asv
dd3d0.m
dd3d1.m
dd3d3.m

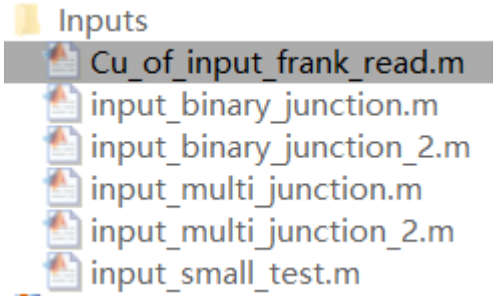


应力应变关系

- **自己修改input文件，添加更多FR源**
- **修改施加应力，应力率；**

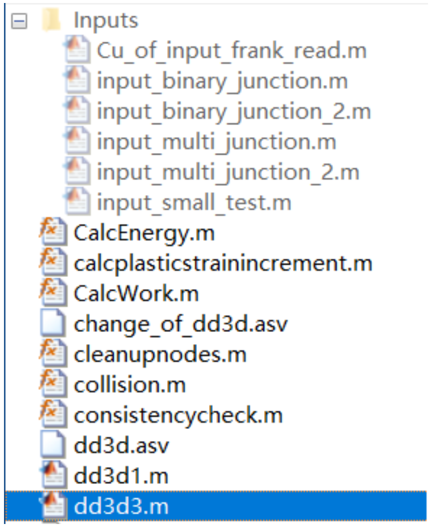
应力率加载的例子

在input文件夹下运行Cu_of_input_frank_read

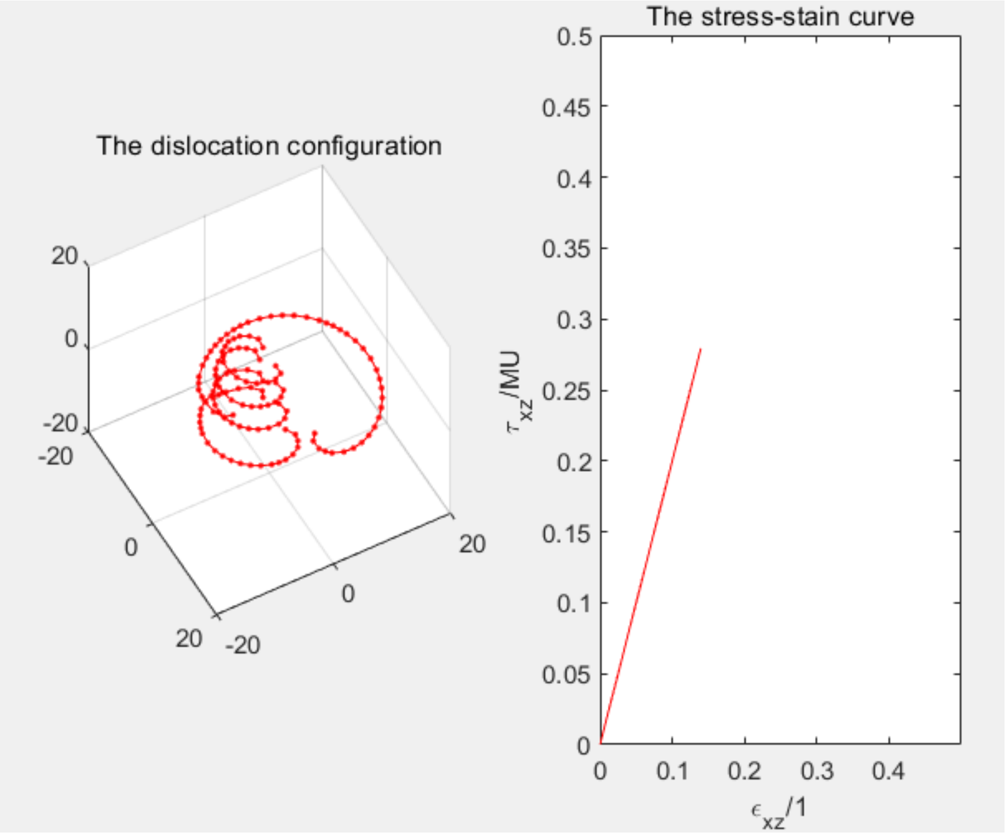


Inputs
Cu_of_input_frank_read.m
input_binary_junction.m
input_binary_junction_2.m
input_multi_junction.m
input_multi_junction_2.m
input_small_test.m

在dd3d_modified文件夹下运行dd3d3



⊟ Inputs
Cu_of_input_frank_read.m
input_binary_junction.m
input_binary_junction_2.m
input_multi_junction.m
input_multi_junction_2.m
input_small_test.m
CalcEnergy.m
calcplasticstrainincrement.m
CalcWork.m
change_of_dd3d.asv
cleanupnodes.m
collision.m
consistencycheck.m
dd3d.asv
dd3d1.m
dd3d3.m


The dislocation configuration
The stress-stain curve
τ_{xz}/MU
ϵ_{xz}/1