Continuation and bifurcation problems in ordinary differential equations: getting practice with AUTO 07P

Tian Yu

GEMS workshop, November 2017

Overview of capabilities

The main algorithms in AUTO are aimed at the continuation of solutions of the dynamical system of the form:

$$u'(t) = f(u(t), p), \qquad f(.,.), u(.) \in R^n,\ p \in R^m \quad (1)$$

$u(t)$: ***state variables*** vector

p : ***parameters*** vector

subject to ***boundary (including initial) conditions*** and **integral constraints**. Above, P denotes one or more free parameters.

The goal is to compute the solutions $u(t)$ changes as P varies.

For the ODE (1) the program can :
Compute families of stable and unstable periodic solutions and **determine the stability along these families**.

Locate folds, branch points, period doubling bifurcations, and bifurcations to tori, along families of periodic solutions. Branch switching is possible at branch points and at period doubling bifurcations.

Follow curves of homoclinic orbits and detect and continue various codimension-2 bifurcations.

Compute solution curves to (1) on **[0,1]**, subject to **general nonlinear boundary and integral conditions**.

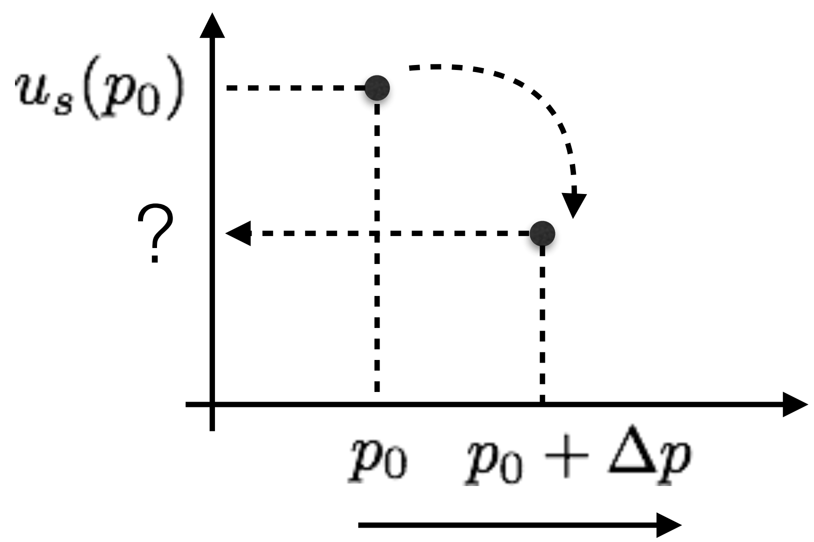
Determine **folds and branch points** along solution families to the above boundary value problem. Branch switching is possible at branch points. Curves of folds and branch points can be computed.

Why a continuation algorithm ?

Given the (stationary) solution $u_s(p_0)$ at $p = p_0$ of the model

$$f(u, p) = 0$$

The goal is to estimate the new stationary point at $p = p_0 + \Delta p$ from the older one.



Continuation algorithms aim to trace the loci of the solutions of algebraic equations given a start solution.

AUTO uses **pseudo-arclength continuation** for following solution families. The pseudo-arclength stepsize is the distance between the current solution and the next solution on a family.

Basic notions and user supplied files

AUTO 07P is constituted by a series of programs in Fortran 90, that are arranged in a series of subdirectories.

SRC: AUTO Source files. They must not be modified by the user

CMDS: AUTO executables

DEMOS: The directory AUTO 07P/demos contains a series of folders each of them is an example. They can be modified by the user, who can adapt them to specific problems.

For each example, two files may be modified by the user:
The Equations-File xxx.90: A Fortran90 file where user will write the **mathematical model (i.e. equations)** to be investigated. **Start solution** is usually included in the equation file at the first run step.

The Constants-file c.xxx: A text file containing the **numerical parameters** used for each continuation

These are the **only** files to be needed during the computation

User supplied files - File xxx.f90

The **mathematical model** is written in the file **xxx.f90**

- It is constituted by different Fortran **subroutines.**

SUBROUTINE FUNC

- In this subroutine one defines the mathematical model, i.e. the ODEs.
- The **state variables** are defined as components of the U vector: **U(1), U(2)……**Depends on the dimension of the problem.
- Parameters are defined as components of the PAR vector: **PAR(1), PAR(2)……**

SUBROUTINE STPNT

- The starting configuration/point for the continuation are here reported (STartingPoiNT)

SUBROUTINE BCND

- Define boundary conditions. To be used when dealing with boundary value problems.

SUBROUTINE ICND

- Definition of Integral CoNDitions

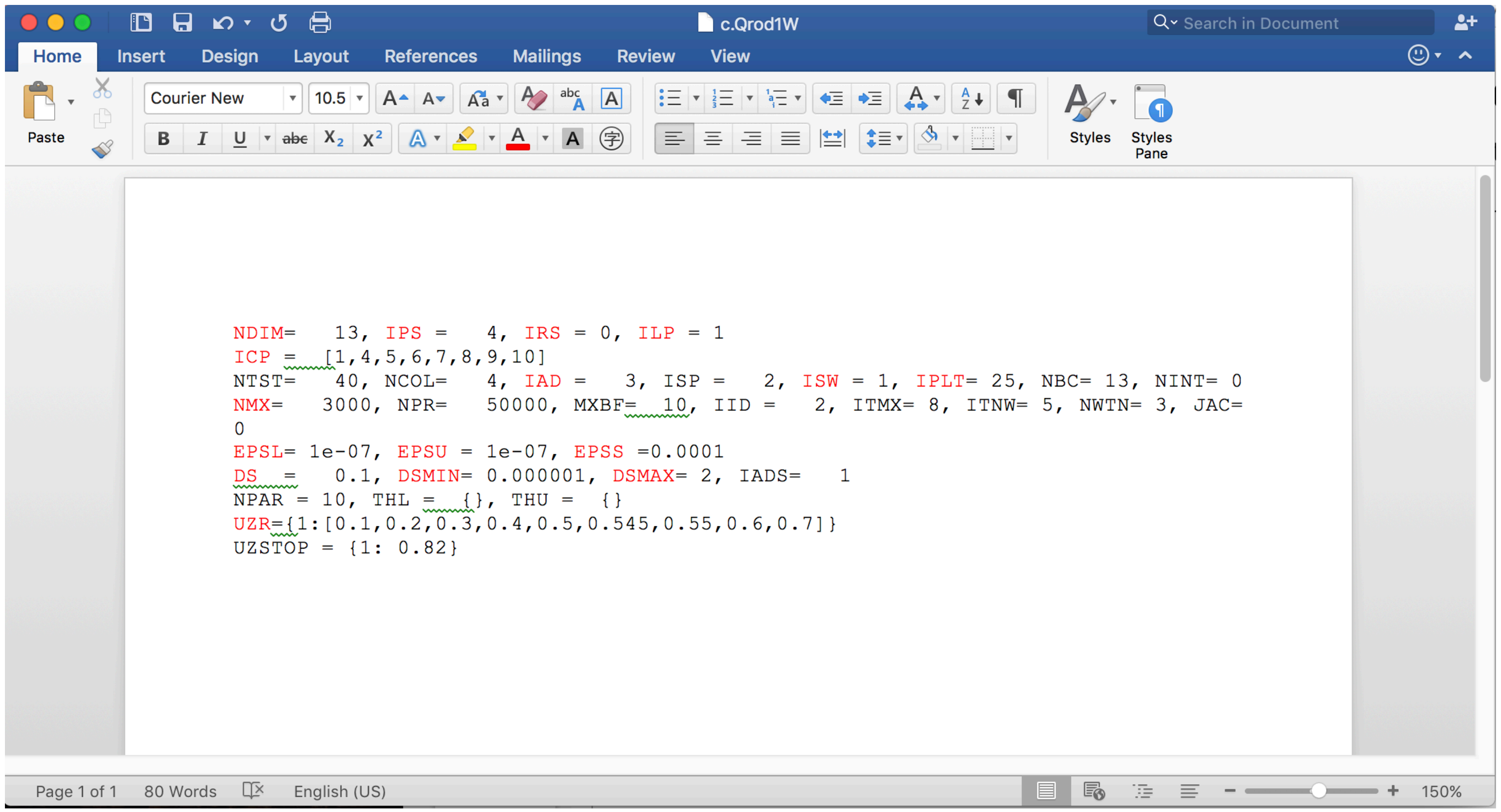
SUBROUTINE PVLS

- Definition of solution measures

User supplied files - File **c.xxx**

After the file xxx.f90 is written, it will be **no longer** modified during the continuation.

Hereafter, the user will manage **only** the constants-file c.xxx for all the simulations.




```
NDIM=   13, IPS =   4, IRS = 0, ILP = 1
ICP =  [1,4,5,6,7,8,9,10]
NTST=   40, NCOL=   4, IAD =   3, ISP =   2, ISW = 1, IPLT= 25, NBC= 13, NINT= 0
NMX=   3000, NPR=   50000, MXBF=  10, IID =   2, ITMX= 8, ITNW= 5, NWTN= 3, JAC=
0
EPSL= 1e-07, EPSU = 1e-07, EPSS =0.0001
DS  =   0.1, DSMIN= 0.000001, DSMAX= 2, IADS=   1
NPAR = 10, THL =  {}, THU =  {}
UZR={1:[0.1,0.2,0.3,0.4,0.5,0.545,0.55,0.6,0.7]}
UZSTOP = {1: 0.82}
```

NDIM

Dimension of the system of ODEs (i.e. the number of state variables)

IPS

Define the problem type

- IPS=0: An algebraic bifurcation problem.
- IPS=1: Stationary solutions of ODEs with detection of Hopf bifurcations.
- IPS=2: Computation of periodic solutions.
- IPS=4: A boundary value problem.
- IPS=5: Algebraic optkmizaiotn.

IRS:

Sets the label of the solution where the computation is to be restarted.

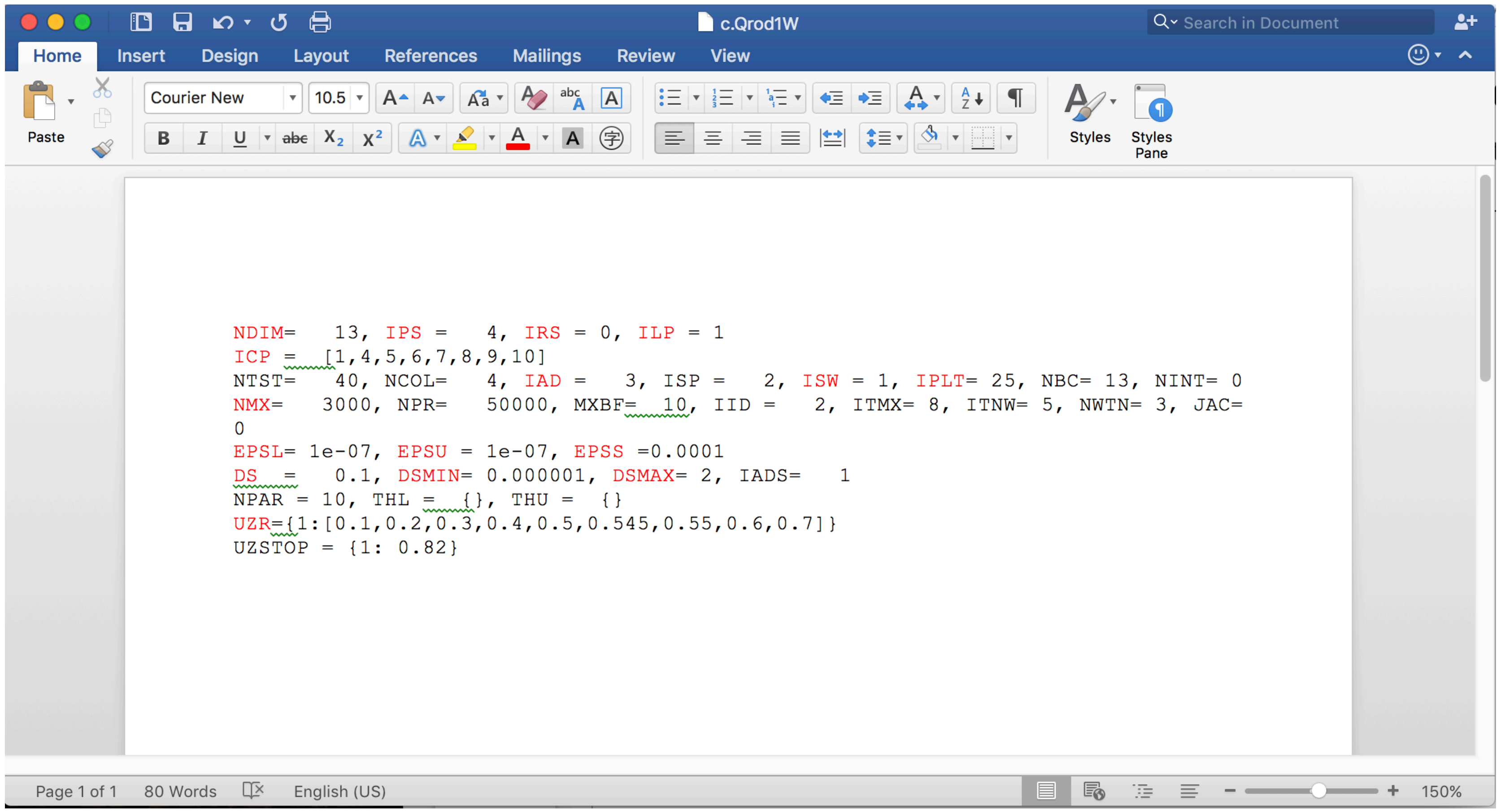
IRS=0: typically used in the first run of a new problem. A starting solution **must be** defined in the user-supplied routine STPNT.

IRS>0: Restart the computation at the previously computed solution with label IRS.

User supplied files - File **c.xxx**

After the file xxx.f90 is written, it will be **no longer** modified during the continuation.

Hereafter, the user will manage **only** the constants-file c.xxx for all the simulations.




```
NDIM=   13, IPS =   4, IRS = 0, ILP = 1
ICP =  [1,4,5,6,7,8,9,10]
NTST=   40, NCOL=   4, IAD =   3, ISP =   2, ISW = 1, IPLT= 25, NBC= 13, NINT= 0
NMX=   3000, NPR=   50000, MXBF=  10, IID =   2, ITMX= 8, ITNW= 5, NWTN= 3, JAC=
0
EPSL= 1e-07, EPSU = 1e-07, EPSS =0.0001
DS  =   0.1, DSMIN= 0.000001, DSMAX= 2, IADS=   1
NPAR = 10, THL =  {}, THU =  {}
UZR={1:[0.1,0.2,0.3,0.4,0.5,0.545,0.55,0.6,0.7]}
UZSTOP = {1: 0.82}
```

NMX:
The maximum number of steps to be taken along any solution branch.

DS
Real number. This constant defines stepsize to be used for the first attempted step along the solution branch. DS may be chosen positive or negative: changing its sign reverses the direction of the computation.

ISW: integer number
ISW=1: normal continuation
ISW=2: two parameters continuation
ISW=-1: To be used when a **"branch-switching" is required.**

ICP:
Array of the free parameters used for the continuation.
The parameters that appears first in the ICP list is called the **"principle continuation parameter"**.

DSMIN, DSMAX, UZR, JAC, IAD, EPSL, EPSU, EPSS, ITMX, NWTN, ITNW and IPLT: See the AUTO document, or use the **values recommended** by AUTO.

AUTO 07P Example - Bratu's Equation **xxx.f90 file**

The goal is to compute a solution family to the boundary value problem

$$u_1' = u_2 \,,$$
$$u_2' = -p_1 e^{u_1} \,.$$

with boundary conditions $u_1(0) = 0, \quad u_1(1) = 0.$

```
SUBROUTINE FUNC(NDIM,U,ICP,PAR,IJAC,F,DFDU,DFDP)
---------- ----

IMPLICIT NONE
INTEGER, INTENT(IN) :: NDIM, ICP(*), IJAC
DOUBLE PRECISION, INTENT(IN) :: U(NDIM), PAR(*)
DOUBLE PRECISION, INTENT(OUT) :: F(NDIM)
DOUBLE PRECISION, INTENT(INOUT) :: DFDU(NDIM,NDIM), DFDP(NDIM,*)

 F(1)= U(2)
 F(2)=-PAR(1) * EXP(U(1))

END SUBROUTINE FUNC
```

```
SUBROUTINE STPNT(NDIM,U,PAR,T)
---------- -----

IMPLICIT NONE
INTEGER, INTENT(IN) :: NDIM
DOUBLE PRECISION, INTENT(INOUT) :: U(NDIM),PAR(*)
DOUBLE PRECISION, INTENT(IN) :: T

 PAR(1)=0

 U(1)=0.0
 U(2)=0.0

END SUBROUTINE STPNT
```

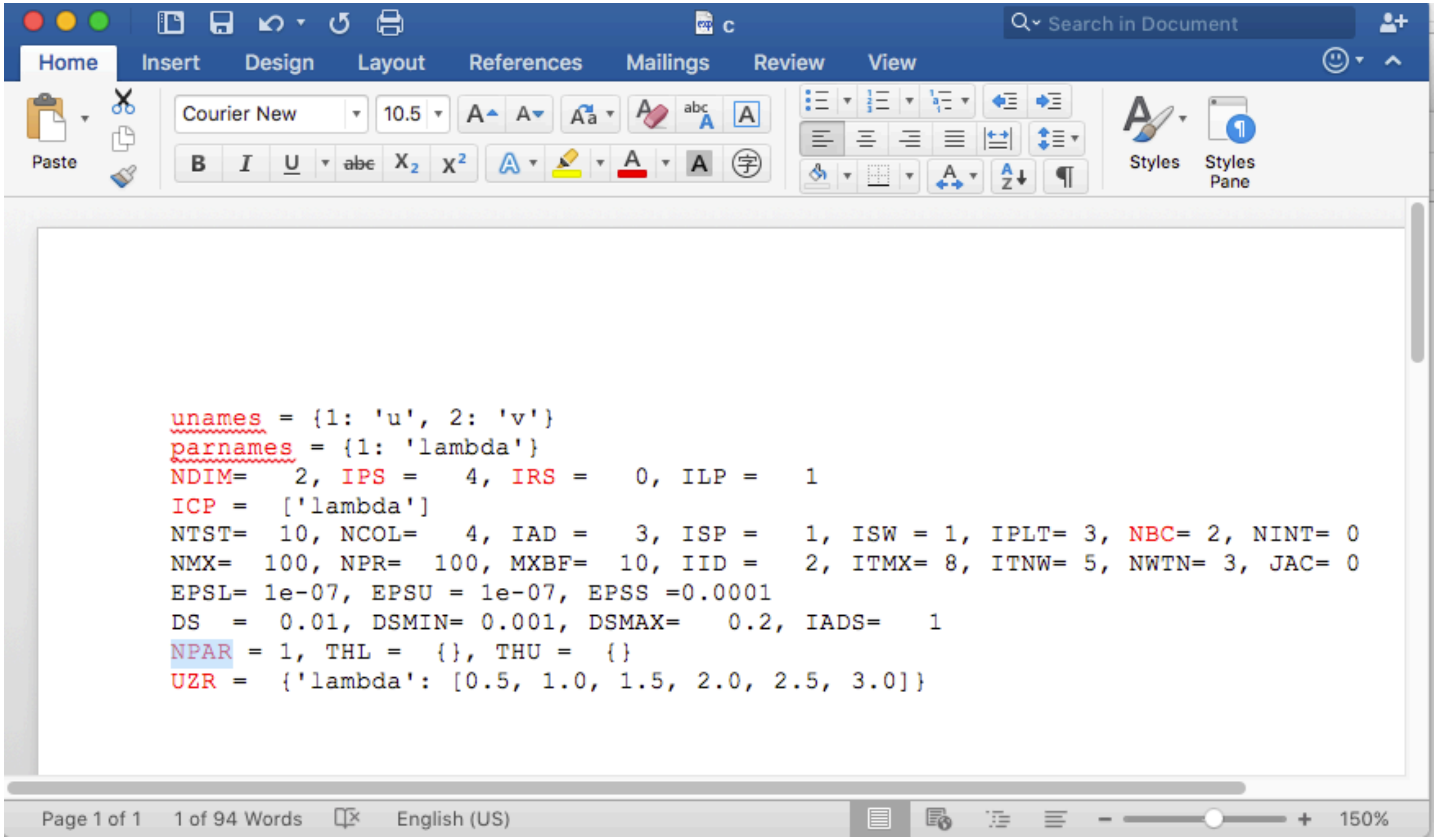
```
SUBROUTINE BCND(NDIM,PAR,ICP,NBC,U0,U1,FB,IJAC,DBC)
---------- ----

IMPLICIT NONE
INTEGER, INTENT(IN) :: NDIM, ICP(*), NBC, IJAC
DOUBLE PRECISION, INTENT(IN) :: PAR(*), U0(NDIM), U1(NDIM)
DOUBLE PRECISION, INTENT(OUT) :: FB(NBC)
DOUBLE PRECISION, INTENT(INOUT) :: DBC(NBC,*)

 FB(1)=U0(1)
 FB(2)=U1(1)

END SUBROUTINE BCND
```

AUTO 07P Example - Bratu's Equation **c.xxx file**

The goal is to compute a solution family to the boundary value problem

$$u_1' = u_2 \,,$$
$$u_2' = -p_1 e^{u_1} \,.$$

with boundary conditions $u_1(0) = 0, \quad u_1(1) = 0.$



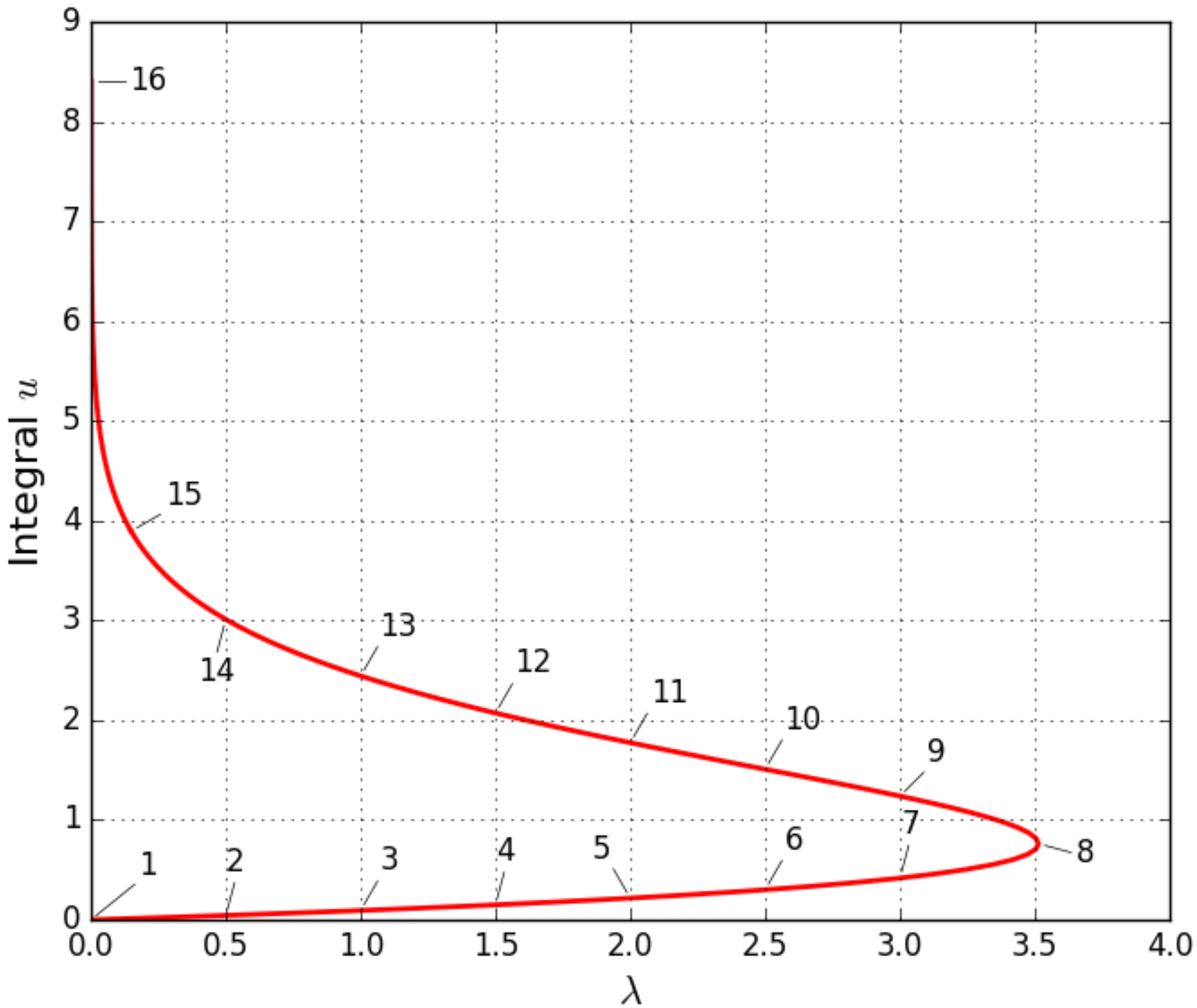
```
unames = {1: 'u', 2: 'v'}
parnames = {1: 'lambda'}
NDIM=   2, IPS =   4, IRS =   0, ILP =   1
ICP =  ['lambda']
NTST=  10, NCOL=   4, IAD =   3, ISP =   1, ISW = 1, IPLT= 3, NBC= 2, NINT= 0
NMX=  100, NPR=  100, MXBF=  10, IID =   2, ITMX= 8, ITNW= 5, NWTN= 3, JAC= 0
EPSL= 1e-07, EPSU = 1e-07, EPSS =0.0001
DS  =  0.01, DSMIN= 0.001, DSMAX=   0.2, IADS=   1
NPAR = 1, THL =  {}, THU =  {}
UZR =  {'lambda': [0.5, 1.0, 1.5, 2.0, 2.5, 3.0]}
```

AUTO 07P Example - Bratu's Equation

The goal is to compute a solution family to the boundary value problem

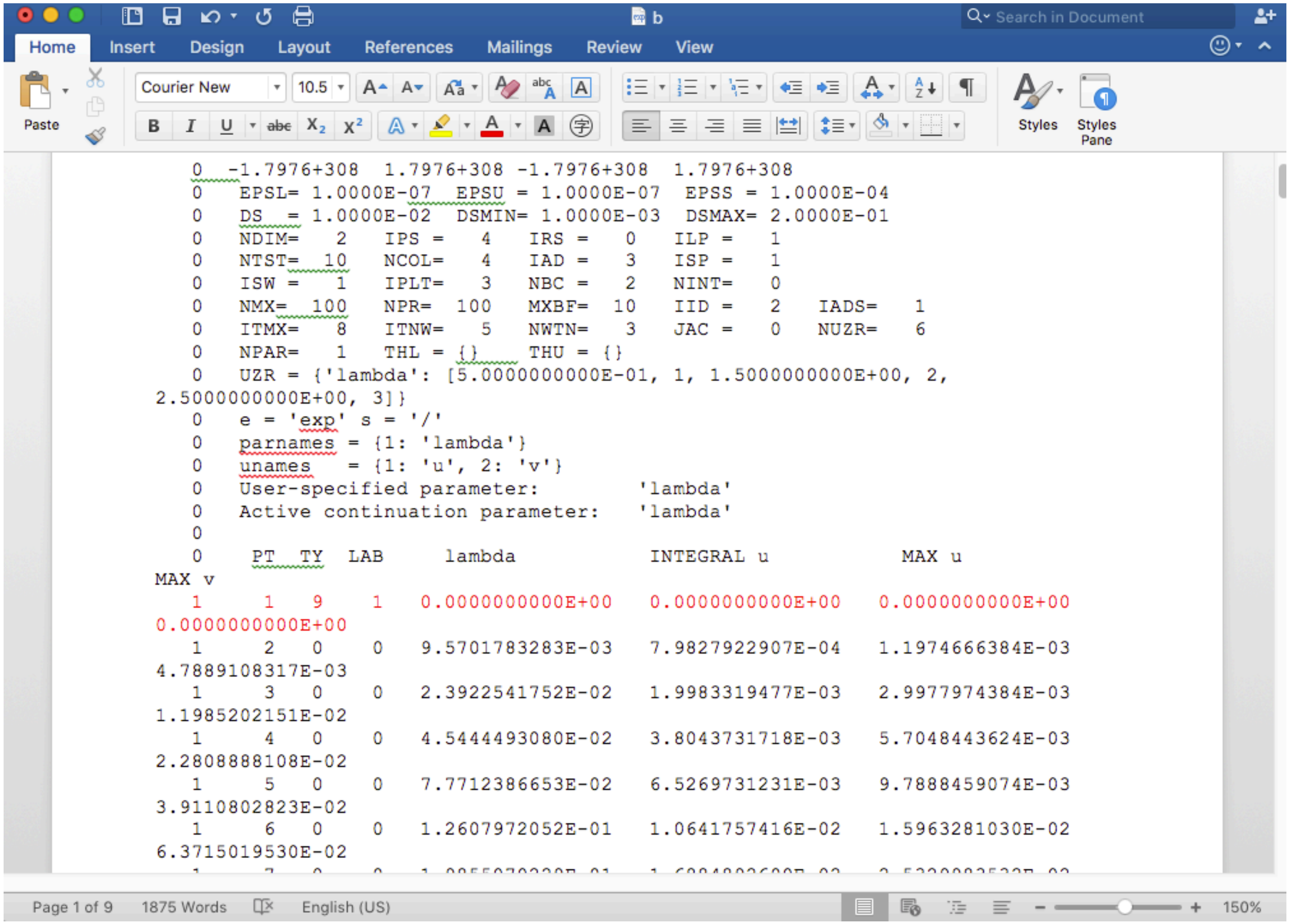
$$u_1' = u_2\,,$$
$$u_2' = -p_1 e^{u_1}\,.$$

with boundary conditions $u_1(0) = 0, \quad u_1(1) = 0.$



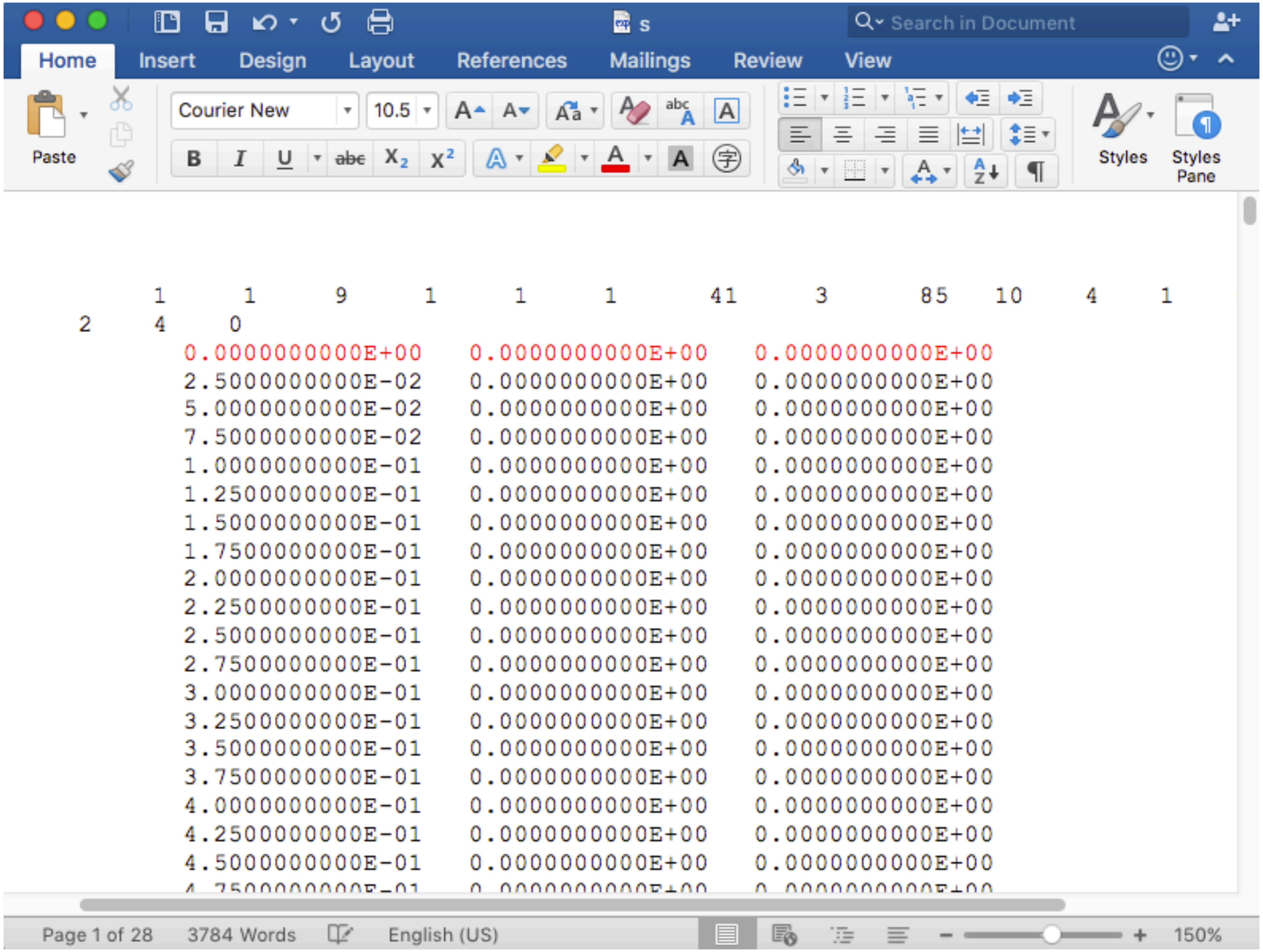

```
print "\n***Compute a solution
family***"
r1=run(e='exp',c='exp')

print "\n***Restart the solution
family***"
r2=run(r1,NTST=20)
save(r1+r2,'exp')
```

Bratu’s Equation - post proces **b.xxx file**



```
   0  -1.7976+308  1.7976+308 -1.7976+308  1.7976+308
   0    EPSL= 1.0000E-07  EPSU = 1.0000E-07  EPSS = 1.0000E-04
   0    DS  = 1.0000E-02  DSMIN= 1.0000E-03  DSMAX= 2.0000E-01
   0    NDIM=    2   IPS =    4   IRS =    0   ILP =    1
   0    NTST=   10   NCOL=    4   IAD =    3   ISP =    1
   0    ISW =    1   IPLT=    3   NBC =    2   NINT=    0
   0    NMX=  100    NPR=  100   MXBF=   10   IID =    2   IADS=    1
   0    ITMX=    8   ITNW=    5   NWTN=    3   JAC =    0   NUZR=    6
   0    NPAR=    1   THL = {}     THU = {}
   0    UZR = {'lambda': [5.0000000000E-01, 1, 1.5000000000E+00, 2,
2.5000000000E+00, 3]}
   0    e = 'exp' s = '/'
   0    parnames = {1: 'lambda'}
   0    unames   = {1: 'u', 2: 'v'}
   0    User-specified parameter:        'lambda'
   0    Active continuation parameter:   'lambda'
   0
   0     PT  TY  LAB     lambda          INTEGRAL u          MAX u
MAX v
   1      1   9    1   0.0000000000E+00   0.0000000000E+00   0.0000000000E+00
0.0000000000E+00
   1      2   0    0   9.5701783283E-03   7.9827922907E-04   1.1974666384E-03
4.7889108317E-03
   1      3   0    0   2.3922541752E-02   1.9983319477E-03   2.9977974384E-03
1.1985202151E-02
   1      4   0    0   4.5444493080E-02   3.8043731718E-03   5.7048443624E-03
2.2808888108E-02
   1      5   0    0   7.7712386653E-02   6.5269731231E-03   9.7888459074E-03
3.9110802823E-02
   1      6   0    0   1.2607972052E-01   1.0641757416E-02   1.5963281030E-02
6.3715019530E-02
```

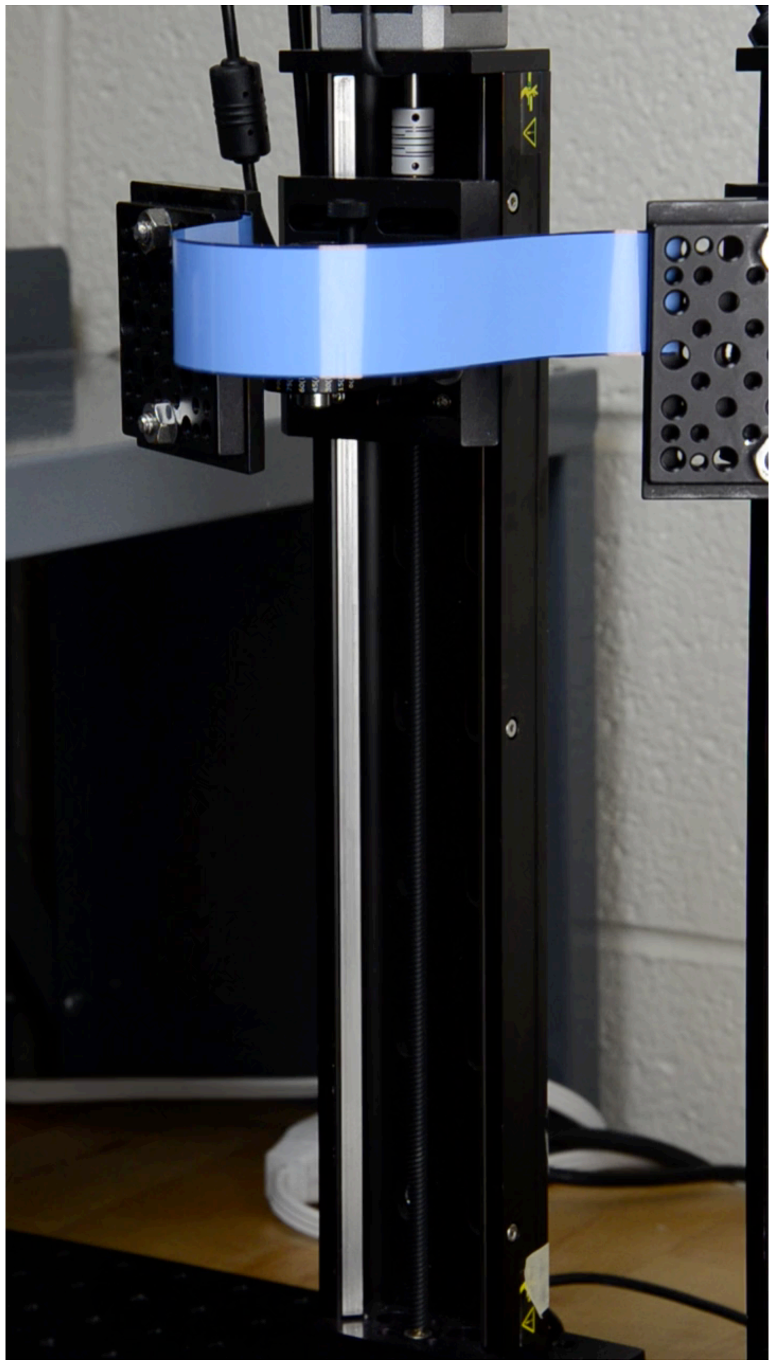
Bratu's Equation - post proces **s.xxx file**




```
          1       1       9       1       1       1       41       3       85    10     4     1
    2     4     0
       0.000000000E+00    0.000000000E+00    0.000000000E+00
       2.500000000E-02    0.000000000E+00    0.000000000E+00
       5.000000000E-02    0.000000000E+00    0.000000000E+00
       7.500000000E-02    0.000000000E+00    0.000000000E+00
       1.000000000E-01    0.000000000E+00    0.000000000E+00
       1.250000000E-01    0.000000000E+00    0.000000000E+00
       1.500000000E-01    0.000000000E+00    0.000000000E+00
       1.750000000E-01    0.000000000E+00    0.000000000E+00
       2.000000000E-01    0.000000000E+00    0.000000000E+00
       2.250000000E-01    0.000000000E+00    0.000000000E+00
       2.500000000E-01    0.000000000E+00    0.000000000E+00
       2.750000000E-01    0.000000000E+00    0.000000000E+00
       3.000000000E-01    0.000000000E+00    0.000000000E+00
       3.250000000E-01    0.000000000E+00    0.000000000E+00
       3.500000000E-01    0.000000000E+00    0.000000000E+00
       3.750000000E-01    0.000000000E+00    0.000000000E+00
       4.000000000E-01    0.000000000E+00    0.000000000E+00
       4.250000000E-01    0.000000000E+00    0.000000000E+00
       4.500000000E-01    0.000000000E+00    0.000000000E+00
       4.750000000E-01    0.000000000E+00    0.000000000E+00
```

Shear a thin band - a two-point boundary value problem

Kirchhoff rod theory



Kirchhoff rod theory

$$\boldsymbol{N}' = \boldsymbol{0}\,,$$
$$\boldsymbol{M}' + \boldsymbol{d}_3 \times \boldsymbol{N} = \boldsymbol{0}\,.$$

$$N_1 = 0, N_2 = 0, N_3 = 0\,,$$
$$\kappa_1 = 2\pi, M_2 = 0, \tau = 0\,,$$
$$q_0 = \tfrac{\sqrt{2}}{2}\sin(\pi s)\,,\ q_1 = -\tfrac{\sqrt{2}}{2}\cos(\pi s)\,,$$
$$q_2 = \tfrac{\sqrt{2}}{2}\sin(\pi s)\,,\ q_3 = \tfrac{\sqrt{2}}{2}\cos(\pi s)\,,$$
$$x = -\tfrac{\sin 2\pi s}{2\pi}\,,\ y = \tfrac{1-\cos 2\pi s}{2\pi}, z = 0\,,$$

$$N_1' = N_2\tau\,,$$
$$N_2' = -N_1\tau + N_3\kappa_1\,,$$
$$N_3' = -N_2\kappa_1\,,$$
$$a\kappa_1' = M_2\tau + N_2\,,$$
$$M_2' = (1-a)\kappa_1\tau - N_1\,,$$
$$\tau' = -M_2\kappa_1\,,$$
$$q_0' = \frac{1}{2}(-q_1\kappa_1 - q_3\tau)\,,\ q_1' = \frac{1}{2}(q_0\kappa_1 + q_2\tau)\,,$$
$$q_2' = \frac{1}{2}(q_3\kappa_1 - q_1\tau)\,,\ q_3' = \frac{1}{2}(-q_2\kappa_1 + q_0\tau)\,,$$
$$x' = 2(q_1q_3 + q_0q_2)\,,\ y' = 2(q_2q_3 - q_0q_1)\,,\ z' = 2(q_0^2 + q_3^2 - \tfrac{1}{2})\,.$$

$$q_0(0) = q_2(0) = q_2(1) = \tfrac{\sqrt{2}}{2}\cos\tfrac{\psi_0}{2}\,,$$
$$q_1(0) = -q_1(1) = -q_3(0) = q_3(1) = -\tfrac{\sqrt{2}}{2}\sin\tfrac{\psi_0}{2}\,,$$
$$x(0) = y(0) = y(1) = z(1) = 0\,,\ x(1) = \Delta L/L\,,\ z(0) = -\Delta D/L\,,$$

Shear a thin band - a two-point boundary value problem

SUBROUTINE FUNC

$$
\begin{aligned}
N_1' &= N_2\tau\,,\\
N_2' &= -N_1\tau + N_3\kappa_1\,,\\
N_3' &= -N_2\kappa_1\,,\\
a\kappa_1' &= M_2\tau + N_2\,,\\
M_2' &= (1-a)\kappa_1\tau - N_1\,,\\
\tau' &= -M_2\kappa_1\,,\\
q_0' &= \frac{1}{2}(-q_1\kappa_1 - q_3\tau)\,,\ q_1' = \frac{1}{2}(q_0\kappa_1 + q_2\tau)\,,\\
q_2' &= \frac{1}{2}(q_3\kappa_1 - q_1\tau)\,,\ q_3' = \frac{1}{2}(-q_2\kappa_1 + q_0\tau)\,,\\
x' &= 2(q_1q_3 + q_0q_2)\,,\ y' = 2(q_2q_3 - q_0q_1)\,,\ z' = 2(q_0^2 + q_3^2 - \tfrac{1}{2})\,.
\end{aligned}
$$

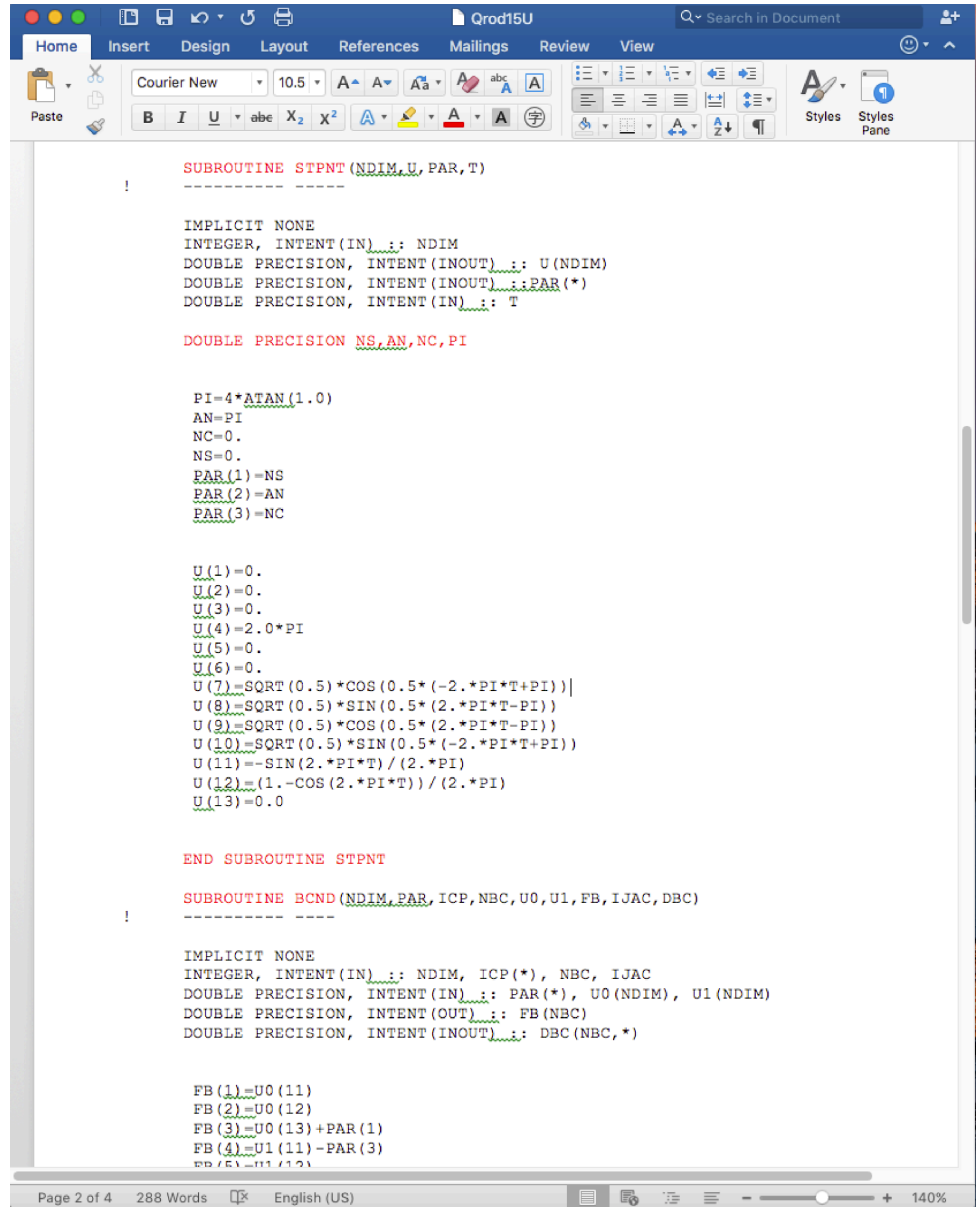
```
!--------------------------------------------------------------------
!   rod : rod model initial guess from circular solution
!   Quaternion description
      SUBROUTINE FUNC(NDIM,U,ICP,PAR,IJAC,F,DFDU,DFDP)
!     ---------- ----

      IMPLICIT NONE
      INTEGER, INTENT(IN) :: NDIM, ICP(*), IJAC
      DOUBLE PRECISION, INTENT(IN) :: U(NDIM), PAR(*)
      DOUBLE PRECISION, INTENT(OUT) :: F(NDIM)
      DOUBLE PRECISION, INTENT(INOUT):: DFDU(NDIM,NDIM)
      DOUBLE PRECISION, INTENT(INOUT):: DFDP(NDIM,*)


DOUBLE PRECISION N1,N2,N3,k1,M,tau,q0,q1,q2,q3
DOUBLE PRECISION Ux,Uy,Uz,a,b,AN,NC,NS


       N1=U(1)
       N2=U(2)
       N3=U(3)
       k1=U(4)
       M=U(5)
       tau=U(6)
       q0=U(7)
       q1=U(8)
       q2=U(9)
       q3=U(10)
       Ux=U(11)
       Uy=U(12)
       Uz=U(13)


       NS=PAR(1)
       AN=PAR(2)
       NC=PAR(3)
       a=0.625

       F(1)=N2*tau
       F(2)=-N1*tau+N3*k1
       F(3)=-N2*k1
       F(4)=(M*tau+N2)/a
       F(5)=((1-a)*tau*k1-N1)
       F(6)=-M*k1
       F(7)=0.5*(-q1*k1-q3*tau)
       F(8)=0.5*(q0*k1+q2*tau)
       F(9)=0.5*(q3*k1-q1*tau)
       F(10)=0.5*(-q2*k1+q0*tau)
       F(11)=2.0*(q1*q3+q0*q2)
       F(12)=2.0*(q2*q3-q0*q1)
       F(13)=2.0*(q0*q0+q3*q3-0.5)

      END SUBROUTINE FUNC
```

Page 1 of 4   288 Words   English (US)   140%

Shear a thin band - a two-point boundary value problem

SUBROUTINE STPNT

$$N_1 = 0, N_2 = 0, N_3 = 0\,,$$
$$\kappa_1 = 2\pi, M_2 = 0, \tau = 0\,,$$
$$q_0 = \frac{\sqrt{2}}{2}\sin(\pi s)\,,\ q_1 = -\frac{\sqrt{2}}{2}\cos(\pi s)\,,$$
$$q_2 = \frac{\sqrt{2}}{2}\sin(\pi s)\,,\ q_3 = \frac{\sqrt{2}}{2}\cos(\pi s)\,,$$
$$x = -\frac{\sin 2\pi s}{2\pi}\,,\ y = \frac{1-\cos 2\pi s}{2\pi}, z = 0\,,$$



```
      SUBROUTINE STPNT(NDIM,U,PAR,T)
!     ---------- -----

      IMPLICIT NONE
      INTEGER, INTENT(IN) :: NDIM
      DOUBLE PRECISION, INTENT(INOUT) :: U(NDIM)
      DOUBLE PRECISION, INTENT(INOUT) ::PAR(*)
      DOUBLE PRECISION, INTENT(IN) :: T

      DOUBLE PRECISION NS,AN,NC,PI


       PI=4*ATAN(1.0)
       AN=PI
       NC=0.
       NS=0.
       PAR(1)=NS
       PAR(2)=AN
       PAR(3)=NC


       U(1)=0.
       U(2)=0.
       U(3)=0.
       U(4)=2.0*PI
       U(5)=0.
       U(6)=0.
       U(7)=SQRT(0.5)*COS(0.5*(-2.*PI*T+PI))
       U(8)=SQRT(0.5)*SIN(0.5*(2.*PI*T-PI))
       U(9)=SQRT(0.5)*COS(0.5*(2.*PI*T-PI))
       U(10)=SQRT(0.5)*SIN(0.5*(-2.*PI*T+PI))
       U(11)=-SIN(2.*PI*T)/(2.*PI)
       U(12)=(1.-COS(2.*PI*T))/(2.*PI)
       U(13)=0.0

      END SUBROUTINE STPNT

      SUBROUTINE BCND(NDIM,PAR,ICP,NBC,U0,U1,FB,IJAC,DBC)
!     ---------- ----

      IMPLICIT NONE
      INTEGER, INTENT(IN) :: NDIM, ICP(*), NBC, IJAC
      DOUBLE PRECISION, INTENT(IN) :: PAR(*), U0(NDIM), U1(NDIM)
      DOUBLE PRECISION, INTENT(OUT) :: FB(NBC)
      DOUBLE PRECISION, INTENT(INOUT) :: DBC(NBC,*)


       FB(1)=U0(11)
       FB(2)=U0(12)
       FB(3)=U0(13)+PAR(1)
       FB(4)=U1(11)-PAR(3)
```

Shear a thin band - a two-point boundary value problem

SUBROUTINE BCND

$$q_0(0) = q_2(0) = q_2(1) = \frac{\sqrt{2}}{2}\cos\frac{\psi_0}{2}\,,$$

$$q_1(0) = -q_1(1) = -q_3(0) = q_3(1) = -\frac{\sqrt{2}}{2}\sin\frac{\psi_0}{2}\,,$$

$$x(0) = y(0) = y(1) = z(1) = 0\,,\ x(1) = \Delta L/L\,,\ z(0) = -\Delta D/L\,,$$

```
DOUBLE PRECISION N1,N2,N3,k1,M,tau,q0,q1,q2,q3
DOUBLE PRECISION Ux,Uy,Uz,a,b,AN,NC,NS


         N1=U(1)
         N2=U(2)
         N3=U(3)
         k1=U(4)
         M=U(5)
         tau=U(6)
         q0=U(7)
         q1=U(8)
         q2=U(9)
         q3=U(10)
         Ux=U(11)
         Uy=U(12)
         Uz=U(13)
```

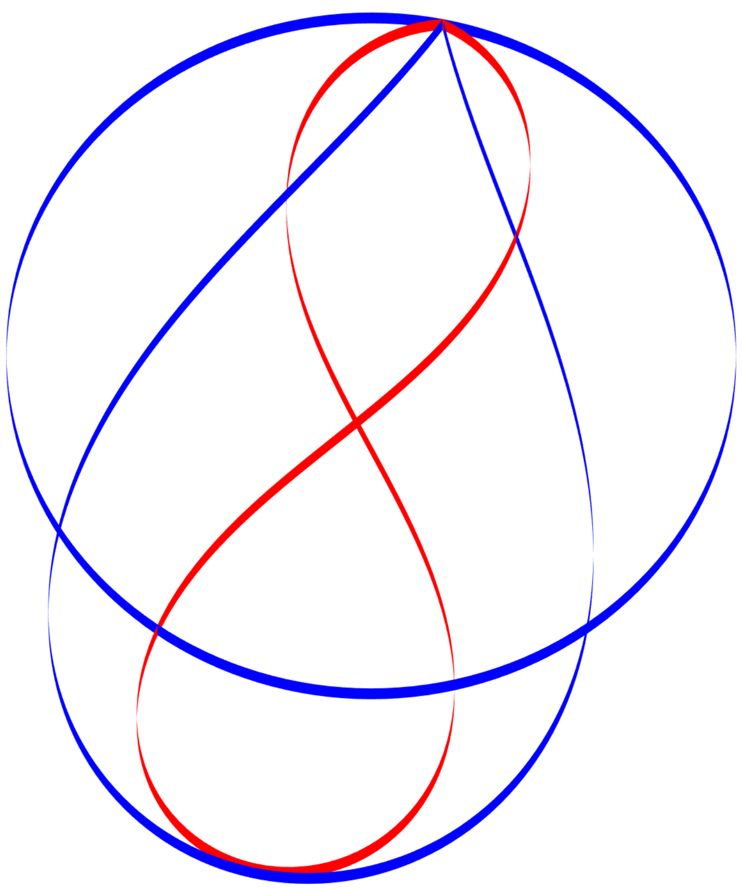
Qrod15U

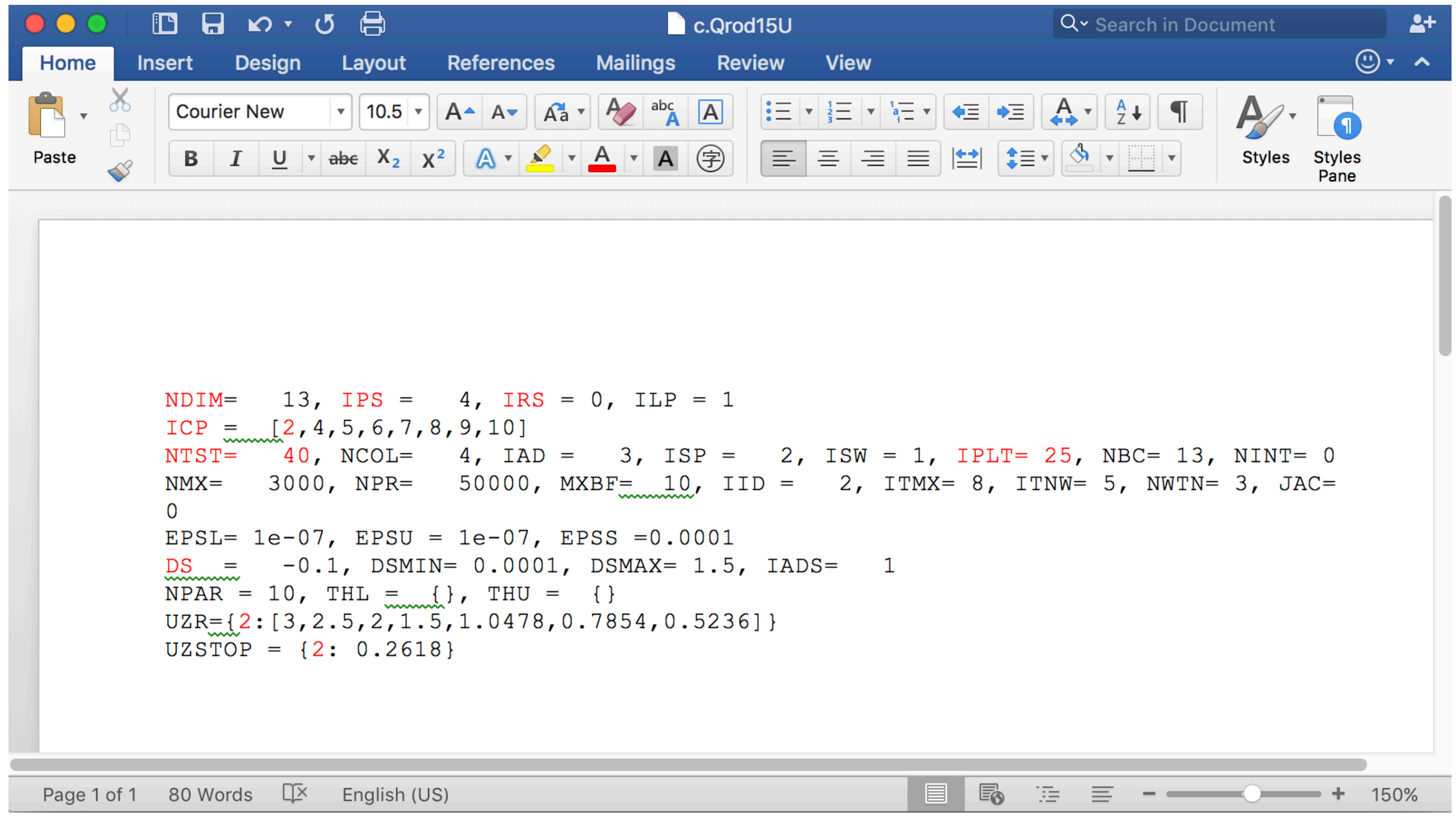
```
      SUBROUTINE BCND(NDIM,PAR,ICP,NBC,U0,U1,FB,IJAC,DBC)
!     ---------- ----

      IMPLICIT NONE
      INTEGER, INTENT(IN) :: NDIM, ICP(*), NBC, IJAC
      DOUBLE PRECISION, INTENT(IN) :: PAR(*), U0(NDIM), U1(NDIM)
      DOUBLE PRECISION, INTENT(OUT) :: FB(NBC)
      DOUBLE PRECISION, INTENT(INOUT) :: DBC(NBC,*)


       FB(1)=U0(11)
       FB(2)=U0(12)
       FB(3)=U0(13)+PAR(1)
       FB(4)=U1(11)-PAR(3)
       FB(5)=U1(12)

       FB(6)=U1(13)
       FB(7)=U0(7)-SQRT(0.5)*COS(PAR(2)*0.5)
       FB(8)=U0(8)+SQRT(0.5)*SIN(PAR(2)*0.5)
       FB(9)=U0(9)-SQRT(0.5)*COS(PAR(2)*0.5)
       FB(10)=U0(10)-SQRT(0.5)*SIN(PAR(2)*0.5)

       FB(11)=U1(10)+SQRT(0.5)*SIN(PAR(2)*0.5)
       FB(12)=U1(8)-SQRT(0.5)*SIN(PAR(2)*0.5)
       FB(13)=U1(9)-SQRT(0.5)*COS(PAR(2)*0.5)


      END SUBROUTINE BCND

      SUBROUTINE ICND
      END SUBROUTINE ICND

      SUBROUTINE FOPT
      END SUBROUTINE FOPT


      SUBROUTINE PVLS(NDIM,U,PAR)
!     ---------- ----
```

Page 3 of 4 288 Words English (US) 140%

Shear a thin band - a two-point boundary value problem

Rotate the two ends

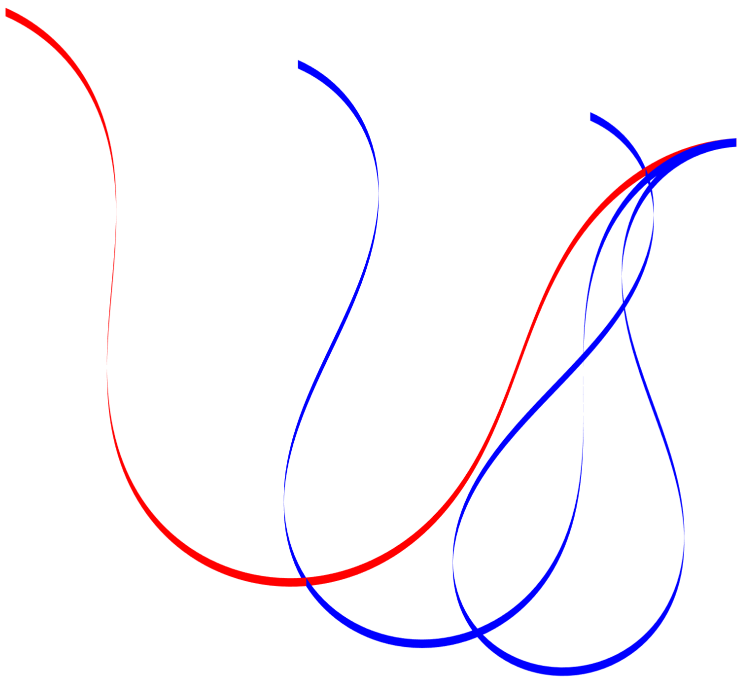


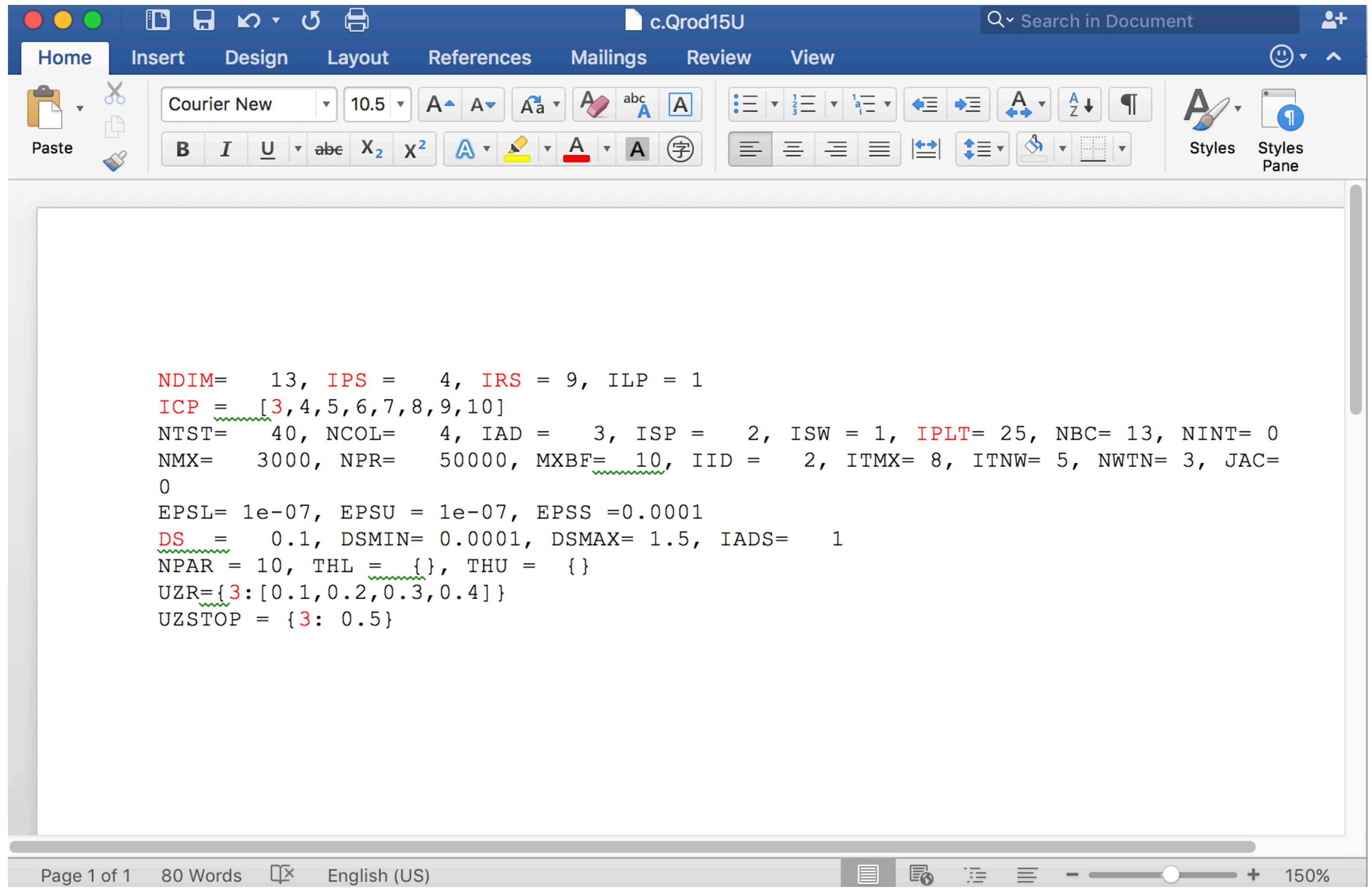


```
NDIM=   13, IPS =   4, IRS = 0, ILP = 1
ICP =  [2,4,5,6,7,8,9,10]
NTST=   40, NCOL=   4, IAD =   3, ISP =   2, ISW = 1, IPLT= 25, NBC= 13, NINT= 0
NMX=   3000, NPR=   50000, MXBF=  10, IID =   2, ITMX= 8, ITNW= 5, NWTN= 3, JAC=
0
EPSL= 1e-07, EPSU = 1e-07, EPSS =0.0001
DS   =   -0.1, DSMIN= 0.0001, DSMAX= 1.5, IADS=   1
NPAR = 10, THL =  {}, THU =  {}
UZR={2:[3,2.5,2,1.5,1.0478,0.7854,0.5236]}
UZSTOP = {2: 0.2618}
```

Shear a thin band - a two-point boundary value problem

Open the two ends

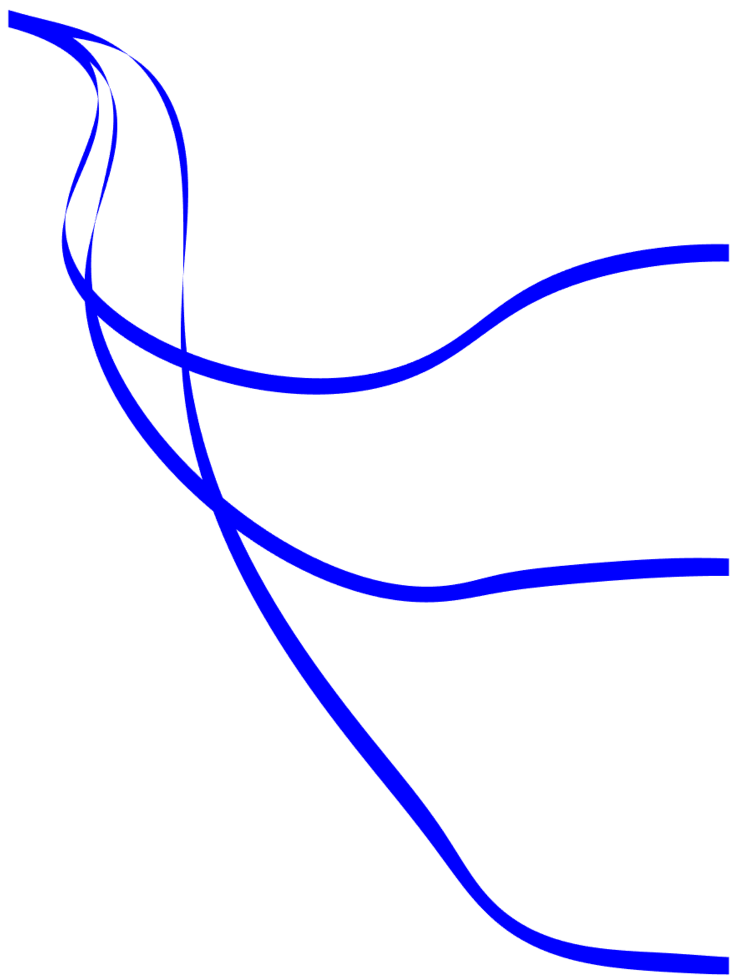





```
NDIM=   13, IPS =   4, IRS = 9, ILP = 1
ICP =  [3,4,5,6,7,8,9,10]
NTST=   40, NCOL=   4, IAD =   3, ISP =   2, ISW = 1, IPLT= 25, NBC= 13, NINT= 0
NMX=   3000, NPR=   50000, MXBF=  10, IID =   2, ITMX= 8, ITNW= 5, NWTN= 3, JAC=
0
EPSL= 1e-07, EPSU = 1e-07, EPSS =0.0001
DS  =   0.1, DSMIN= 0.0001, DSMAX= 1.5, IADS=   1
NPAR = 10, THL =  {}, THU =  {}
UZR={3:[0.1,0.2,0.3,0.4]}
UZSTOP = {3: 0.5}
```

Shear a thin band - a two-point boundary value problem

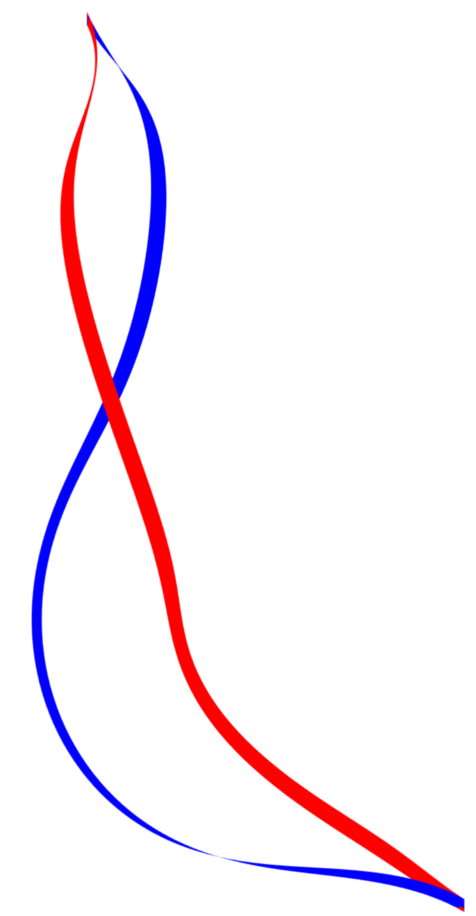
Apply shear

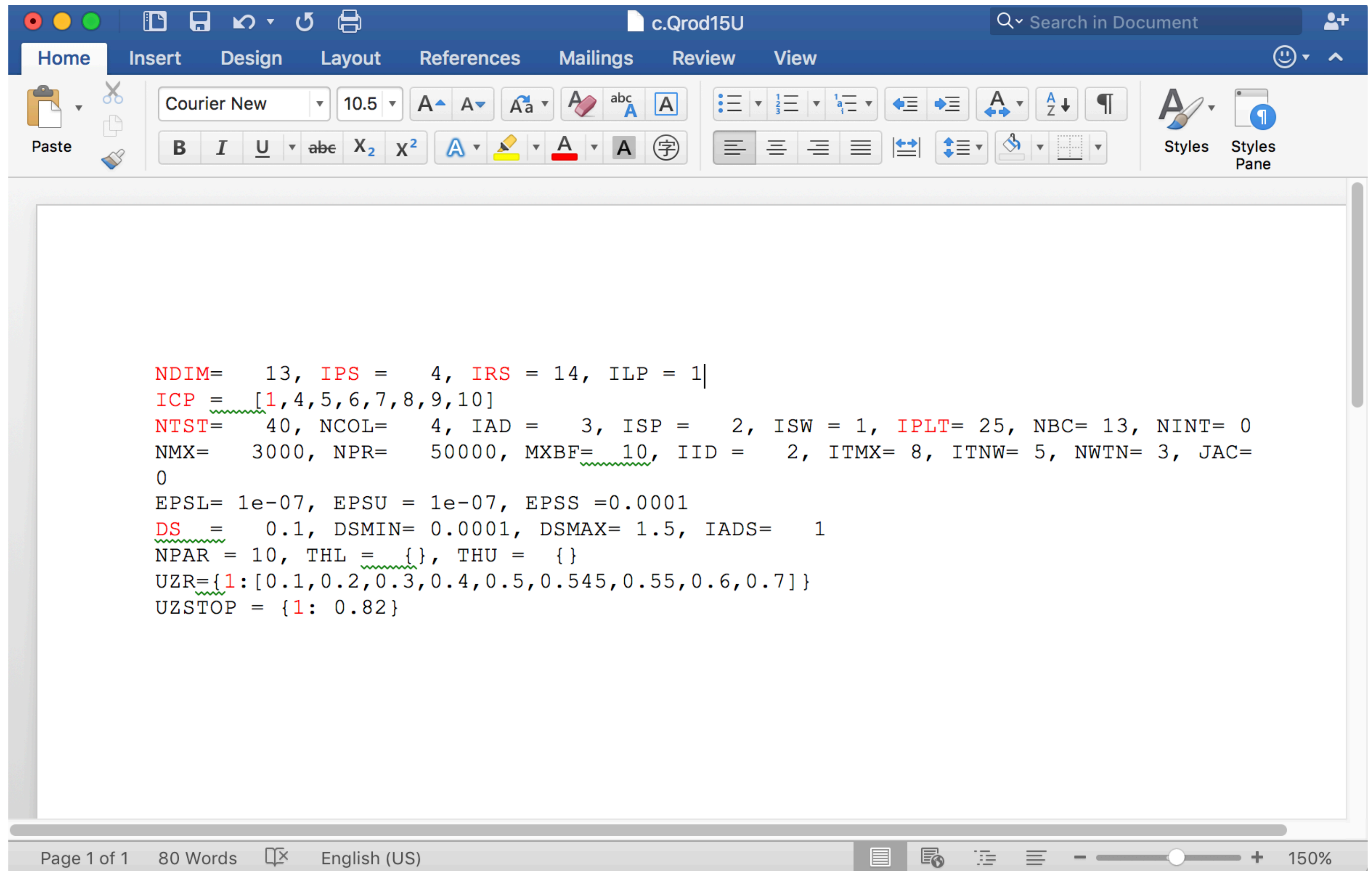


```
NDIM=   13, IPS =   4, IRS = 14, ILP = 1
ICP =  [1,4,5,6,7,8,9,10]
NTST=   40, NCOL=   4, IAD =   3, ISP =   2, ISW = 1, IPLT= 25, NBC= 13, NINT= 0
NMX=   3000, NPR=   50000, MXBF=  10, IID =   2, ITMX= 8, ITNW= 5, NWTN= 3, JAC=
0
EPSL= 1e-07, EPSU = 1e-07, EPSS =0.0001
DS  =   0.1, DSMIN= 0.0001, DSMAX= 1.5, IADS=   1
NPAR = 10, THL =  {}, THU =  {}
UZR={1:[0.1,0.2,0.3,0.4,0.5,0.545,0.55,0.6,0.7]}
UZSTOP = {1: 0.82}
```

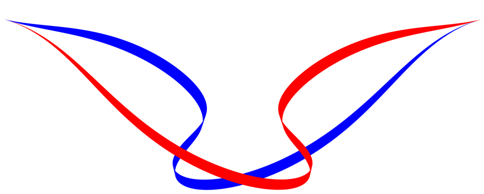
Shear a thin band - a two-point boundary value problem

Pitchfork bifurcation



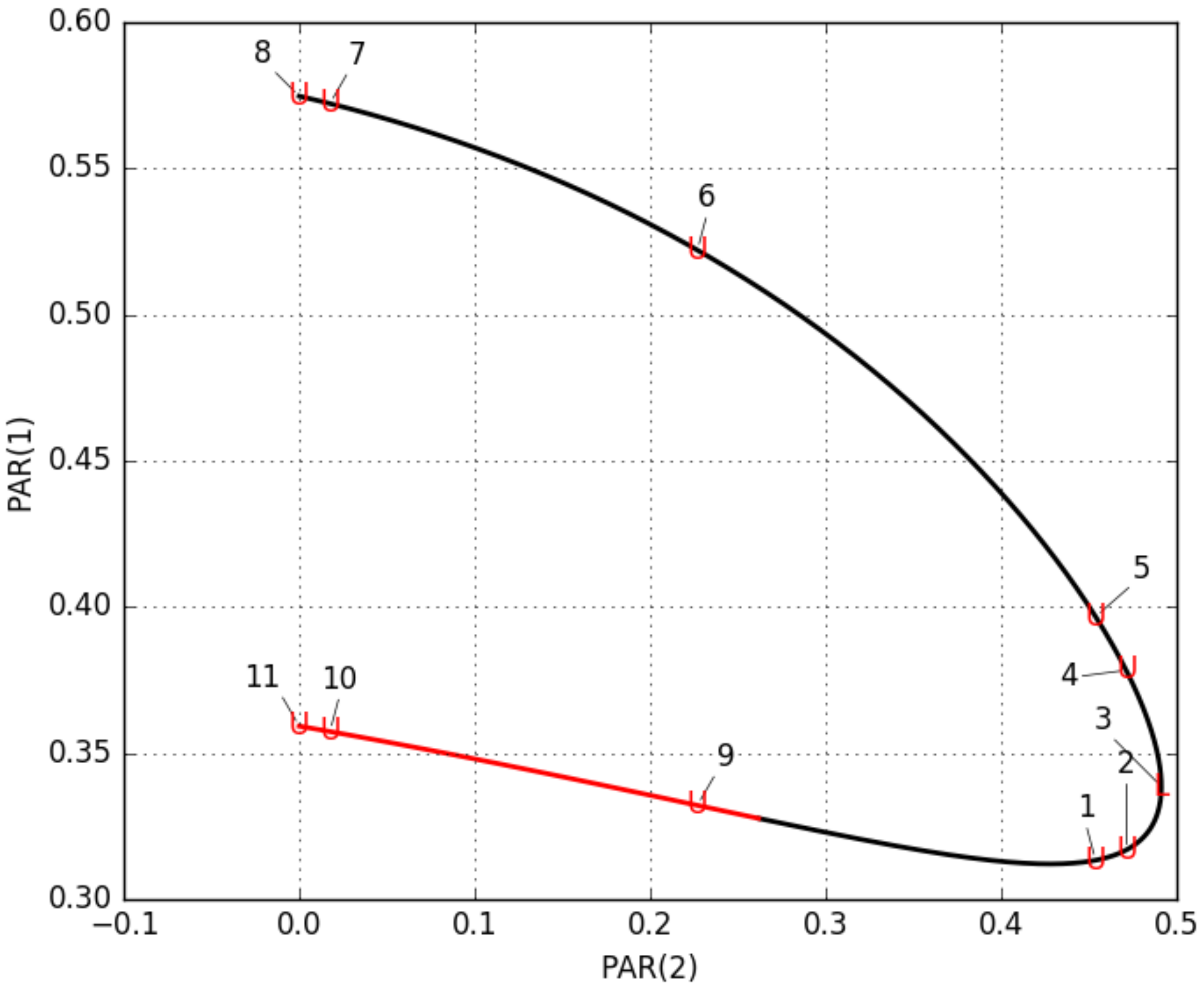


```
NDIM=   13, IPS =   4, IRS = 14, ILP = 1
ICP =  [1,4,5,6,7,8,9,10]
NTST=   40, NCOL=   4, IAD =   3, ISP =   2, ISW = 1, IPLT= 25, NBC= 13, NINT= 0
NMX=   3000, NPR=   50000, MXBF=  10, IID =   2, ITMX= 8, ITNW= 5, NWTN= 3, JAC=
0
EPSL= 1e-07, EPSU = 1e-07, EPSS =0.0001
DS  =   0.1, DSMIN= 0.0001, DSMAX= 1.5, IADS=   1
NPAR = 10, THL =  {}, THU =  {}
UZR={1:[0.1,0.2,0.3,0.4,0.5,0.545,0.55,0.6,0.7]}
UZSTOP = {1: 0.82}
```



Shear a thin band - a two-point boundary value problem

Two parameter continuation



print "\n***two parameter continuation***"
print "\n***generate starting data for the loci of the first BP***"
run15UBP1=run(run15Ushear('BP1'),ICP=[2,1],ISW=2)
save(run15UBP1,'15UBP1')

print "\n*** Compute the loci of BP1 forward***"
run15UBP1locif=run(run15UBP1,DSMAX=0.1,NMX=2000, DS=0.1,UZR={2:[0.0175,0.2269,0.4538,0.4712]},UZSTOP = {2: 0})
save(run15BP1locif,'15BP1locif')

print "\n*** Compute the loci of BP1 backward***"
run15UBP1locib=run(run15UBP1,DSMAX=0.1,NMX=2000, DS=-0.1,UZR={2: [0.0175,0.2269,0.4538,0.4712,0.7854]},UZSTOP = {2: 0})
save(run15BP1locib,'15BP1locib')

run15UBP1loci=run15UBP1locif+run15UBP1locib
run15UBP1loci=rl(run15UBP1loci)
save(run15UBP1loci,'15BP1loci')