```
clear;clf; % Program ColP3D.m compatible with MATLAB R2008a and later
% Run Parameters that can be set by the user
% penetration flag 0=PKN, 1=P3D, dimensionless: toughness, leak-off;
  ipen = 1;                                        K = 1;     C = 1;
% # of Colloc pts, Geometric time factor,  Max Newton iters,    residual Tol
  N = 10;          rfac = 2;               Maxit = 20;          tol=1e-9;
% some constants and default parameters
gamma_0=1.0006328466775270;gamma_mt=0;t0=1e-6;tM=50;tc=tM;
if C>0,
 gamma_mt=2/pi/C;Cp=10/3;tc=(gamma_mt/gamma_0)^Cp;t0=1e-6/C^Cp;tM=1e5/C^Cp;
end
%
lambda_u    = (8+sqrt(K^4+64))/K^2;  % runaway height growth boundary
% set up mesh and time steps
t=t0;dt=t0/10;dx=1/N;x=0:dx:1;xf=0:dx/20:1;xft=1-dx:dx/50:1;
in=1:N;xi=x(in);xh=0.5*(x(1:N)+x(2:N+1));[xph,is]= sort([x xh(1:N-1)]);
% precalculate PHI(OMEGA)
[PHI,LAMBDA,OMEGA]=PHIOMEGA(N,K,lambda_u,ipen);
% initialize the solution to the storage PKN values
[Omega0,P0,lambda0,Psi0,gamma0,gammadot]=init_PKN(x,t0);% collocation pts
[Omega0h,P0h,lambda0h,Psi0h,gamma0,gammadot]=init_PKN(xh,t0);% 1/2 pts
% now initialize the collocation soln at channel points
y0=[Omega0(in);Psi0(in)];y=y0;y0h=[Omega0h(in);Psi0h(in)];lambda=ones(N+1,1);
itime=0;tauh=0:t/10:t;gammah=gamma_0*tauh.^(4/5);gamma=gamma_0*(t+dt).^(4/5);
xi_l       = 0;
while t < tM & lambda(1)<min(5,lambda_u),
 t=t + dt;if t >= tM,break;end;R=[];tauh=[tauh t];gammah=[gammah gamma];
 for iter = 1:Maxit
  b=getRHS(xi,y,gamma,K,t,tc,y0,y0h,gamma0,dt,OMEGA,PHI,C,tauh,gammah,ipen);
  J= getJacN(xi,y,gamma,K,t,tc,y0,y0h,gamma0,dt,b,OMEGA,PHI,C,tauh,gammah,ipen);
  dS=J\b;y=y-reshape(dS(1:2*N),2,N);gamma=gamma-dS(2*N+1);gammah(end)=gamma;
  Omega=[y(1,:) 0];lambda=LamEval(OMEGA,LAMBDA,Omega);
  if lambda(1)>lambda_u;[' lambda_u exceeded '],break;break;end
  Psi=[y(2,:) 0];if iter >2 & norm(b,2) < tol,break;end
 end
 if iter==Maxit,[' No convergence: Residual = ',num2str(R(iter))],end
 [yh,Vol,Leak,Atau,alp] = getyh(xi,y,gamma,K,t,tc,y0,y0h,gamma0,dt,...
   OMEGA,PHI,C,tauh,gammah,ipen);
 lambda=LamEval(OMEGA,LAMBDA,Omega);itime=itime+1;TS(itime)=t;GAMS(itime)=gamma;
 ip=find(lambda>1,1,'last');
 if ip>1,xi_l = min(x(ip-1)+(1-lambda(ip-1))/(lambda(ip)-lambda(ip-1))*...
      (x(ip)-x(ip-1)),1);end
 XIL(itime)             = xi_l;
 plotsol(N,x,Omega,xf,t,K,C,Psi,ipen,xft,Atau,alp,lambda,lambda_u,...
   TS,GAMS,gamma_0,gamma_mt,XIL)
  % copy vectors over for the next time step
  y0=y;y0h=yh;dgamma=gamma-gamma0;gamma0=gamma;gamma=gamma0+dgamma;
  dt= rfac*dt;
end
```

```
function [PHI,LAMBDA,OMEGA]=PHIOMEGA(N,K,lambda_u,ipen)
% set up form factor PHI a priori  at sample points ready for spline interp
if ipen==0,
 OMEGA=(1:N);LAMBDA=ones(1,N);PHI= ones(1,N);
else
 Omegafun = inline('(k*x.^(3/2)+2*sqrt(x.^2-1))/pi','x','k');
 if K==0,lambdaM=10; else lambdaM=lambda_u;end
 Ns=1500;ns=1:Ns;a=1.11;ep=(exp(lambdaM/Ns/a)-1);
 LAMBDA=[1:0.001:1.1 a*(1+ep).^ns];OMEGA=Omegafun(LAMBDA,K);PHI=Phi(LAMBDA,K);
end
end

function y=Phi(lambda,K)
y=ones(1,length(lambda));i2=find(lambda>1);lam =lambda(i2);
y(i2)=pi^2*(4*sqrt(lam)-K*sqrt(lam.^2-1)).*IntOmega3(lam,K)./(lam.^2.*...
(4*sqrt(lam)+3*K*sqrt(lam.^2-1))./(((K*lam.^(3/2)+2*sqrt(lam.^2-1))/pi).^3)/12;
end

function y = IntOmega3(lambda,K)
for i=1:length(lambda)
 lam=lambda(i);
 y(i)=2*quadgk(@(x)OmegaF3(x,lam,K),0,0.5*lam,'abstol',1e-12,'reltol',1e-8);
end
end

function Omega3 = OmegaF3(x,lambda,K)
Omega3 = ((4/pi)*(((K/pi/sqrt(lambda)-(2/pi)*asin(1/lambda))*...
  sqrt(lambda^2-4*x.^2))+(2/pi)*(sqrt(lambda^2-4*x.^2)*asin(1/lambda)+...
  phi12(x,lambda)))).^3;
end

function y=phi12(x,lambda)
i0=find(abs(x)==0.5);i1=find(abs(x)<0.5);
i2=find(0.5<abs(x)&abs(x)<=0.5*lambda);y(i0)=log(lambda);
y(i1) = -2*x(i1).*atanh((2*x(i1)*sqrt(lambda^2-1))./...
sqrt(lambda^2-4*x(i1).^2))+atanh((sqrt(lambda^2-1))./sqrt(lambda^2-4*x(i1).^2));
y(i2)=-2*x(i2).*atanh(sqrt(lambda^2-4*x(i2).^2)./(2*x(i2)*sqrt(lambda^2-1)))...
 +atanh(sqrt(lambda^2-4*x(i2).^2)./sqrt(lambda^2-1));
end

function [Omega0,P0,lambda0,Psi0,gamma0,gammadot]=init_PKN(x,t0)
% initialize the solution at collocation points to the impermeable PKN
gamma_0=1.0006328466775270;Omega_0=((12/5)*gamma_0^2)^(1/3);
N=length(x);gamma0=gamma_0*t0^(4/5);gammadot=(4/5)*gamma_0*t0^(-1/5);
Omega0=Omega_0*t0^(1/5)*(1-x).^(1/3).*(1-(1-x)/96 +23*(1-x).^2/64512);
P0=Omega0;lambda0=ones(N,1);Psi0=(1-x).^(1/3);
end
```

```
function b = getRHS(x,y,gamma,K,t,tc,y0,y0h,gamma0,dt,OMEGA,PHI,C,tauh,gammah,ipen)
[n,N]= size(y);
F  = FS(x,y,gamma,K,y0,gamma0,dt,OMEGA,PHI,C,tauh,gammah,ipen);
h=diff(x);H=spdiags(h',0,N-1,N-1);ii=1:N-1;ip1=2:N;xi=x(ii);yi=y(:,ii);
Fi=F(:,ii);xip1=x(ip1);yip1=y(:,ip1);Fip1=F(:,ip1);xih=0.5*(xi+xip1);
yih=0.5*(yi+yip1)-(Fip1-Fi)*H/8;
Fih=FS(xih,yih,gamma,K,y0h(:,ii),gamma0,dt,OMEGA,PHI,C,tauh,gammah,ipen);
Phim=yip1-yi-(Fip1+4*Fih+Fi)*H/6;dgamma=(gamma-gamma0)/dt;
%
if t<tc,alp=1/3;Atau=(3*dgamma*gamma)^(1/3);
else alp= 3/8;Atau=2*(C/3)^(1/4)*gamma^(3/8)*dgamma^(1/8);end
Voltip=Atau*h(1)^(1+alp)/(1+alp);tipVal=Atau*h(1)^alp;
Vol=(yip1(1,:)+4*yih(1,:)+yi(1,:))*h'/6+Voltip;
Leaktip=(2/3)*(gamma*dt/(gamma-gamma0))^(1/2)*h(1)^(3/2);
Leak=(sqrt(t-spline(gammah,tauh,gamma*xip1))+...
  4*sqrt(t-spline(gammah,tauh,gamma*xih))+...
  sqrt(t-spline(gammah,tauh,gamma*xi)))*h'/6+Leaktip;
Phig=gamma*(Vol+2*C*Leak)-t;Phib=GS(y(:,1),y(:,N),tipVal);
b=[Phib(:);Phim(:);Phig];
end

function F = FS(x,y,gamma,K,y0,gamma0,dt,OMEGA,PHI,C,tauh,gammah,ipen)
Ups=Upsilon(OMEGA,PHI,y(1,:),ipen);tau=tauh(end);
F(1,:) =-gamma*y(2,:)./y(1,:).^3./Ups;tau0=spline(gammah,tauh,gamma*x);
F(2,:) =-gamma*(y(1,:)-y0(1,:))/dt-(gamma-gamma0)*...
  gamma*x.*y(2,:)./y(1,:).^3./Ups/dt-C*gamma./sqrt(tau-tau0);
end

function Ups = Upsilon(OMEGA,PHI,Omega,ipen)
if ipen==0,Ups=ones(1,length(Omega));
else i1=find(Omega<=OMEGA(1));i2=find(Omega>OMEGA(1));
  Ups(i1)=ones(1,length(i1));Ups(i2)=spline(OMEGA,PHI,Omega(i2));end
%
end

function r=GS(ya,yb,tipVal)
r(1)=ya(2)-1;r(2)=yb(1)-tipVal;end

function J = getJacN(x,y,gamma,K,t,tc,y0,y0h,gamma0,dt,b,OMEGA,PHI,C,...
  tauh,gammah,ipen)
ep=sqrt(eps);[n,N]=size(y);I=eye(n*N);NC=1:(n*N+1);
for j=1:n*N
 yt=reshape(y(:)+ep*I(:,j),n,N);J(NC,j) = (getRHS(x,yt,gamma,K,t,tc,...
   y0,y0h,gamma0,dt,OMEGA,PHI,C,tauh,gammah,ipen)-b)/ep;
end
J(NC,n*N+1) = (getRHS(x,y,gamma+ep,K,t,tc,y0,y0h,gamma0,dt,OMEGA,PHI,C,...
  tauh,[gammah(1:end-1) gammah(end)+ep],ipen)-b)/ep;
end

function lambda = LamEval(OMEGA,LAMBDA,Omega)
i1=find(Omega<=OMEGA(1));i2=find(Omega>OMEGA(1));
lambda(i1)=ones(1,length(i1));lambda(i2) = spline(OMEGA,LAMBDA,Omega(i2));
```

```
end


function [yih,Vol,Leak,Atau,alp] = getyh(x,y,gamma,K,t,tc,y0,y0h,gamma0,...
  dt,OMEGA,PHI,C,tauh,gammah,ipen)
[n,N]=size(y);
F = FS(x,y,gamma,K,y0,gamma0,dt,OMEGA,PHI,C,tauh,gammah,ipen);
h=diff(x);H=spdiags(h',0,N-1,N-1);ii=1:N-1;ip1=2:N;xi=x(ii);yi=y(:,ii);
Fi=F(:,ii);xip1=x(ip1);yip1=y(:,ip1);Fip1=F(:,ip1);xih =0.5*(xi+xip1);
yih=0.5*(yi+yip1)-(Fip1-Fi)*H/8;dgamma=(gamma-gamma0)/dt;
if t<tc,alp=1/3;Atau=(3*dgamma*gamma)^(1/3);
else alp= 3/8;Atau=2*(C/3)^(1/4)*gamma^(3/8)*dgamma^(1/8);end
Voltip=Atau*h(1)^(1+alp)/(1+alp);tipVal= Atau*h(1)^alp;
Vol=(yip1(1,:)+4*yih(1,:)+yi(1,:))*h'/6+Voltip;
Leaktip=(2/3)*(gamma*dt/(gamma-gamma0))^(1/2)*h(1)^(3/2);
Leak=(sqrt(t-spline(gammah,tauh,gamma*xip1))+...
  4*sqrt(t-spline(gammah,tauh,gamma*xih))+...
  sqrt(t-spline(gammah,tauh,gamma*xi)))*h'/6+Leaktip;
end

function plotsol(N,x,Omega,xf,t,K,C,Psi,ipen,xft,Atau,alp,lambda,...
  lambda_u,TS,GAMS,gamma_0,gamma_mt,XIL)
figure(1); subplot(2,1,1);plot(x(1:end-1),Omega(1:end-1),'k-o',...
  xf,exact_PKNC(xf,t,C),'b-',xf,exact_PKN(xf,t),'r-',...
  x,Psi,'ko--',x,ones(1,N+1)*K/pi/ipen,'m',xft,Atau*(1-xft).^alp,'k-',...
  'linewidth',2,'markerfacecolor','k')
xlabel('\xi');ylabel(' \Omega(\xi,\tau) and \Psi(\xi,\tau) ');
tit=([' t = ',num2str(t)]);title(tit);
legend(' \Omega Collocation',' \Omega PKN Permeable ',...
  ' \Omega PKN Impermeable ',' \Psi Collocation ',' \Omega_b ',1)
subplot(2,1,2);plot(x,lambda,'ko-',x,ones(1,N+1)*lambda_u,'r',...
  'linewidth',2,'markerfacecolor','k','markersize',6)
xlabel('\xi');ylabel(' \lambda(\xi) ')
lamx =[' \lambda_u = ',num2str(lambda_u)];
legend(' Collocation',lamx)
if lambda(1)<lambda_u,ax=axis;ax(3)=1;ax(4)=1.2*lambda(1);axis(ax);end
figure(2);subplot(2,1,1);loglog(TS,GAMS,'k-',TS,gamma_0*TS.^(4/5),'r-',...
  TS,gamma_mt*TS.^(1/2),'b-','linewidth',2)
xlabel('\tau');ylabel(' \gamma(\tau) ')
legend(' Collocation P3D ',' Impermeable PKN ',' Permeable PKN ',4)
subplot(2,1,2);semilogx(TS,XIL,'ko','markerfacecolor','k','markersize',6)
ax=axis;ax(3)=0;axis(ax);
xlabel('\tau');ylabel(' \xi_l(\tau) ')
title(' Penetration free boundary ')
pause(0.1)
end

function Omega=exact_PKN(x,t)
gamma_0 = 1.0006328466775270;Omega_0 = ((12/5)*gamma_0^2)^(1/3);
Omega   = Omega_0*t^(1/5)*(1-x).^(1/3).*(1-(1-x)/96 +23*(1-x).^2/64512);
end

function Omega=exact_PKNC(x,t,C)
gamma_m0 = 2/pi/C; Omega_m0 = 2^(11/8)/pi^(1/2)/(3*C)^(1/4);
```

```
Omega     = Omega_m0*t^(1/8)*(1-x).^(3/8).*(1+(1-x)/80);
end
```