NUCLEAR REACTOR KINETICS

Lecture 5

Basic Transient Solutions
to the 1-D, 2-group Diffusion Equations

Massachusetts
Institute of
Technology

Course Outline

22.213 (22.S904) Calendar					
Lecture #	Date	Topic	LECTURER	Read	Assignment Handed Out
1	5-Sep	Course Overview and First Day Exam	Smith		
2	10-Sep	Review of Delayed Neutrons and Point Kinetics Equations	Smith		PSET # 1: Point Kinetics
3	12-Sep	Review Steady-State Finite-Difference Diffusion Methods (1D, 2D)	Smith		
4	17-Sep	Generalized PKEs from Spatial Finite-Difference Diffusion	Smith		PSET # 2: 2-D Steady-State Diffusion
5	19-Sep	Basic Transient Finite-Difference Direct Solutions	Smith		
6	24-Sep	Testing Various PKE implementations	Smith		PSET # 3: 2-D Fully-Implicit Diffusion
7	26-Sep	Quasi-Static Time-Integration	Smith		
8	1-Oct	Higher-order Time Integration	Smith		PSET # 4: PKE from 2-D Diffusion
9	3-Oct	Time Stepping for Automatic Error Control	Smith		
	8-Oct	Columbus Holiday (8th and 9th)			
10	10-Oct	Smith on Travel: Time for Course Project and Homework	None		PSET # 5: PKE Time Step Control
11	15-Oct	Iterative Numerical MethodsL PJ, GS, SOR, CG, GMRES, etc.	Smith		
12	17-Oct	Coarse Mesh Rebalance & Nonlinear Diffusion Acceleration	Smith		PSET # 6: PKE with Nonlinear Feedback
13	22-Oct	Nodal Methods: Kinetic Distortion and Frequency Transformation	Smith		
	24-Oct	Midterm Exam			
14	29-Oct	Midterm Detailed Exam Solution/2D LRA SS Comparisons	Smith		PSET # 7: CMR and NDA acceleration
15	31-Oct	Multigrid Acceleration Methods	Smith		
16	5-Nov	JFNK for Non-linear Systems	Smith		2-D LRA Rod Ejection Problem: Contest
17	7-Nov	Transient Sn	Smith		
	12-Nov	Veterns Day Holiday			
	14-Nov	Special Project Work Period	ANS Meeting		
18	19-Nov	Transient MOC	Smith		
19	21-Nov	Parallel Solver Technologies (PetSc)	Herman/Roberts		
20	26-Nov	So You Want To Be A Professor? Student Lectures	?????		
21	28-Nov	So You Want To Be A Professor? Student Lectures	?????		
22	3-Dec	So You Want To Be A Professor? Student Lectures	?????		
23	5-Dec	So You Want To Be A Professor? Student Lectures	?????		
24	10-Dec	So You Want To Be A Professor? Student Lectures	?????		
25	12-Dec	Last Day of Class General Wrapup, Cats and Dogs, Critique	Smith		
	17-21 Dec	Finals Week - No Exam for 22.S904 (22.213)			

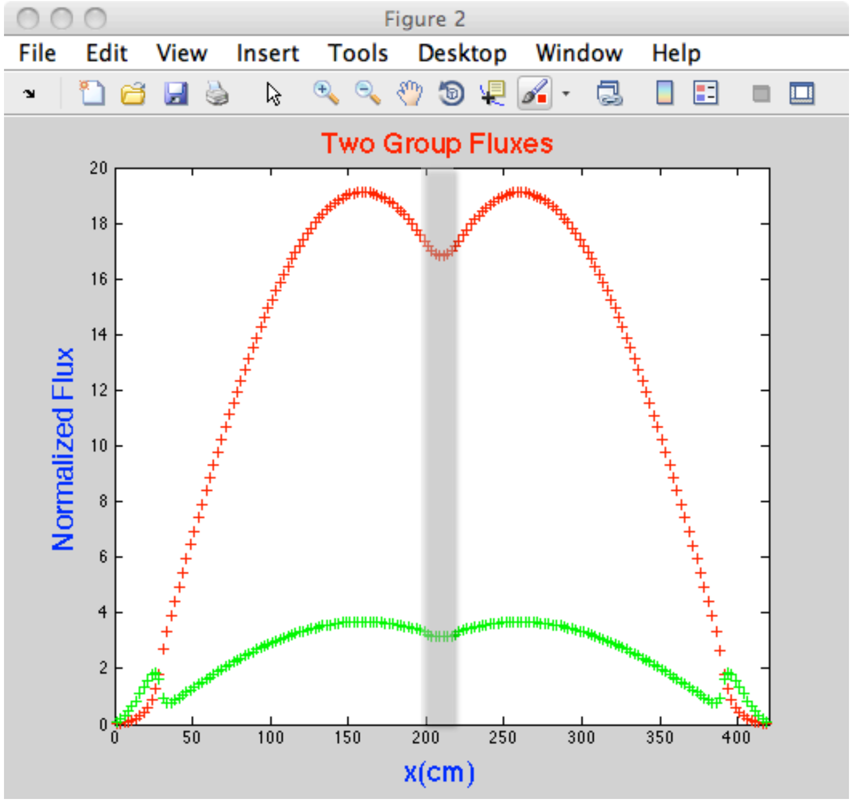
PSET # 2
SOLVING 1-D, 2-GROUP FINITE-DIFFERENCE EQUATIONS

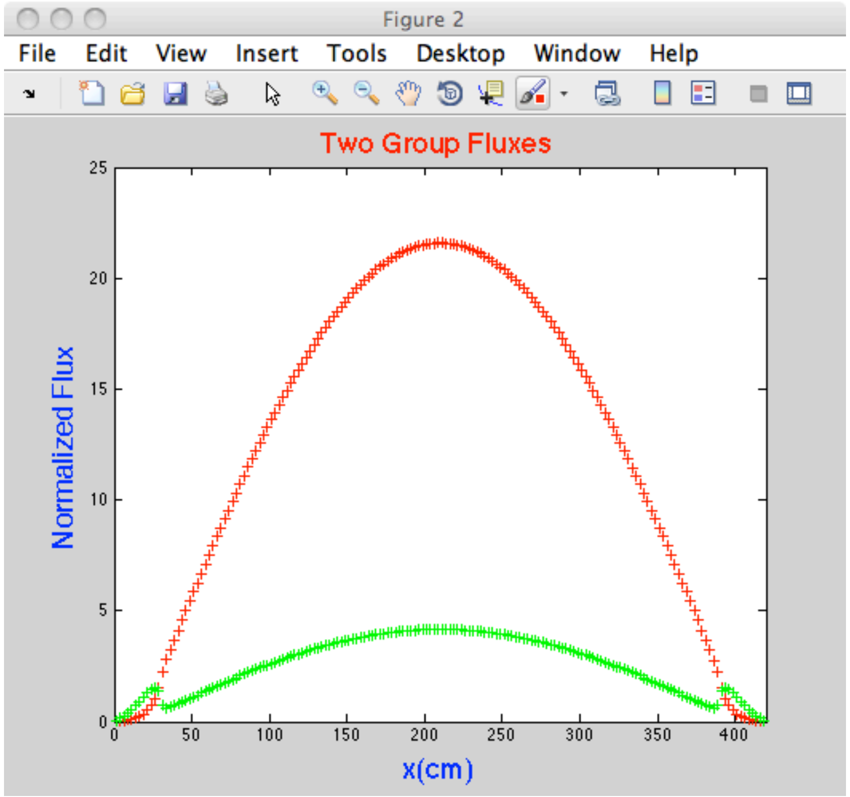
Due: Sept 24, 2012

Massachusetts Institute of Technology

Pset 2: Starting Point for Transient Calculations: Steady-State Solutions

Transient Control Rod Withdrawal in a 1-D Reflected Core: Beginning/Ending Shapes





Simulations in 1-D:

Static Solutions
Dynamic Solutions
Point-Kinetics
Generalized Point-Kinetics
Improved Quasi-Static Methods

Pset 2

Solve the Rodded and Unrodded, 1-D, 2-Group Diffusion Problems for the Following Materials, Geometry, and b.c.s

D1, D2, Sigma-a1, Sigma-a2, Sig1→2, Nu*sigma-f1, Nu*sigma-f2

```
%
% define two group cross sections for all potential materials
%
  nmat=5;
  xs(1,1)=1.300; xs(2,1)=0.500; xs(3,1)=0.0098; xs(4,1)=0.114; xs(5,1)=0.022; xs(6,1)=0.006; xs(7,1)=0.1950;  % 3% enriched fuel
  xs(1,2)=1.300; xs(2,2)=0.500; xs(3,2)=0.0105; xs(4,2)=0.134; xs(5,2)=0.022; xs(6,2)=0.008; xs(7,2)=0.2380;  % 4% enriched fuel
  xs(1,3)=1.500; xs(2,3)=0.500; xs(3,3)=0.0002; xs(4,3)=0.010; xs(5,3)=0.032; xs(6,3)=0.000; xs(7,3)=0.0000;  % water
  xs(1,4)=1.300; xs(2,4)=0.500; xs(3,4)=999.99; xs(4,4)=999.9; xs(5,4)=0.020; xs(6,4)=0.000; xs(7,4)=0.0000;  % black absorber
  xs(1,5)=1.300; xs(2,5)=0.500; xs(3,5)=0.0098; xs(4,5)=0.118; xs(5,5)=0.022; xs(6,5)=0.006; xs(7,5)=0.1950;  % 3% enriched + rod
%
% define problem geometry and assign materials:
%
%   #zones=NZONE w(nzone),n(nzone), mat(nzone)
%
%  bc | slab 1| slab 2| slab 3| ..... slab(NZONE) | bc
%

  NZONE=5;                                          % numer of material zones
  w(1)= 30; w(2)=170; w(3)=20;  w(4)=170; w(5)=30;  % width    per zone
  n(1)= 3;  n(2)=17;  n(3)=2;   n(4)=17;  n(5)=3;   % mesh     per zone
  m(1)= 3;  m(2)=1 ;  m(3)=5;   m(4)=1;   m(5)=3;   % material per zone
  n=n*4;                                            % mesh refinement factor

  bc=2;          % 0=zero flux, 1=zero incoming, 2=reflective bc
```

1. b.c. | Reflector | Fuel-1 | Rod | Fuel-1| Reflector | b.c.
2. Cross sections, zone widths, and cross section given above
3. Reflective b.c. on outer surfaces (you will see why later)

Pset 2

Write your own diffusion solver (in any language you choose) using power iteration with P-J and G-S iterative flux inversion

PART A: Difference Equations

- Derive the expression for the first-order finite-difference net current at a nodal interface for the case of variable mesh spacing/material properties.

PART B: Spatial Convergence

- Plot iteratively-converged eigenvalue and L_2 norm of nodal power error (using 10 cm nodes) vs. mesh spacing until the L_2 norm of error is converged to < 1.e-6 for the rodded and unrodded cores.

PART C: Dominance Ratios

- Plot the asymptotic dominance ratio vs. mesh spacing for the rodded and unrodded cores.

Pset 2

PART D: Iterative Convergence of P-J

- Plot the number of fission source iterations needed to achieve L_2 norm of changes of nodal powers for successive fission source iterations < 1.e-6 vs. flux iteration point-wise L_2 norm for flux convergence criteria of 1.e-1, 1.e-2, 1.e-3, 1.e-4, and 1.e-5 for the rodded and unrodded cores.

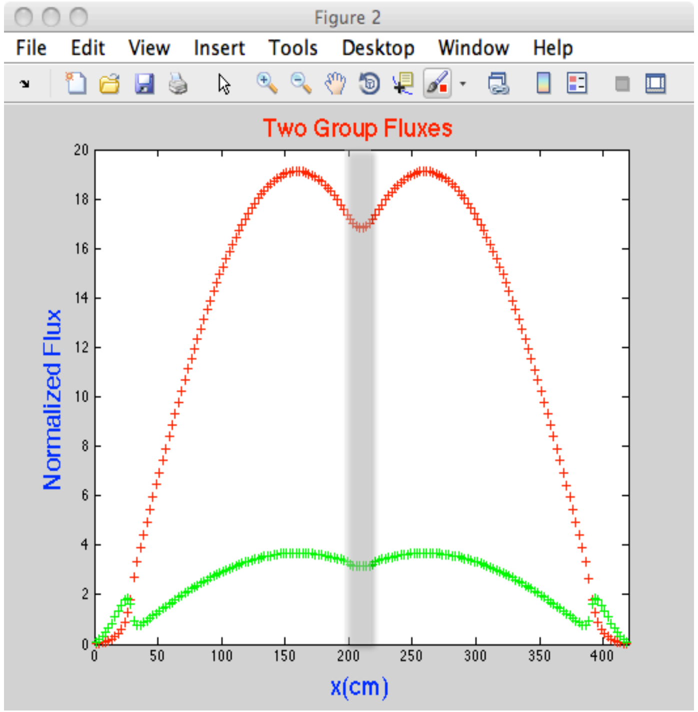
PART E: Iterative Convergence of G-S

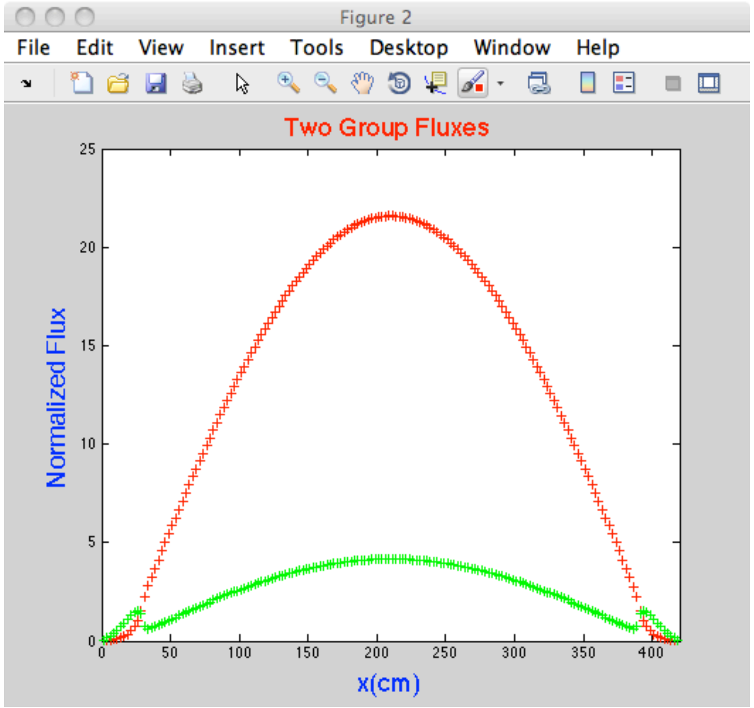
- Plot the number of fission source iterations needed to achieve L_2 norm of changes of nodal powers for successive fission source iterations < 1.e-6 vs. flux iteration point-wise L_2 norm for flux convergence criteria of 1.e-1, 1.e-2, 1.e-3, 1.e-4, and 1.e-5 for the rodded and unrodded cores.

PART F: Real vs. Adjoint Fluxes

1. What are the spatially and iteratively converged real and adjoint eigenvalues for the rodded and unrodded problems?
2. What is the static rod worth in pcm?
3. Plot the spatially and iteratively converged real and adjoint fluxes for the rodded and unrodded problems.

Background Reading





DIFFUSION THEORY METHODS FOR SPATIAL KINETICS CALCULATIONS

T. M. Sutton and B. N. Aviles

Knolls Atomic Power Laboratory, P. O. Box 1072, Schenectady, NY 12301-1072, U.S.A.

Today's Lecture: Goals

- Questions about Pset 2 assignment:1-D, 2-group finite-difference
- Pset 3 assignment:1-D, 2-group, transient finite-difference solutions
- Look at 2 very different transients (delayed and super-prompt critical)
- Examine IKE edits
- Understand PKE form spatial solutions
- Understand nuances of various PKE approaches
- Discuss upcoming lectures and paths forward

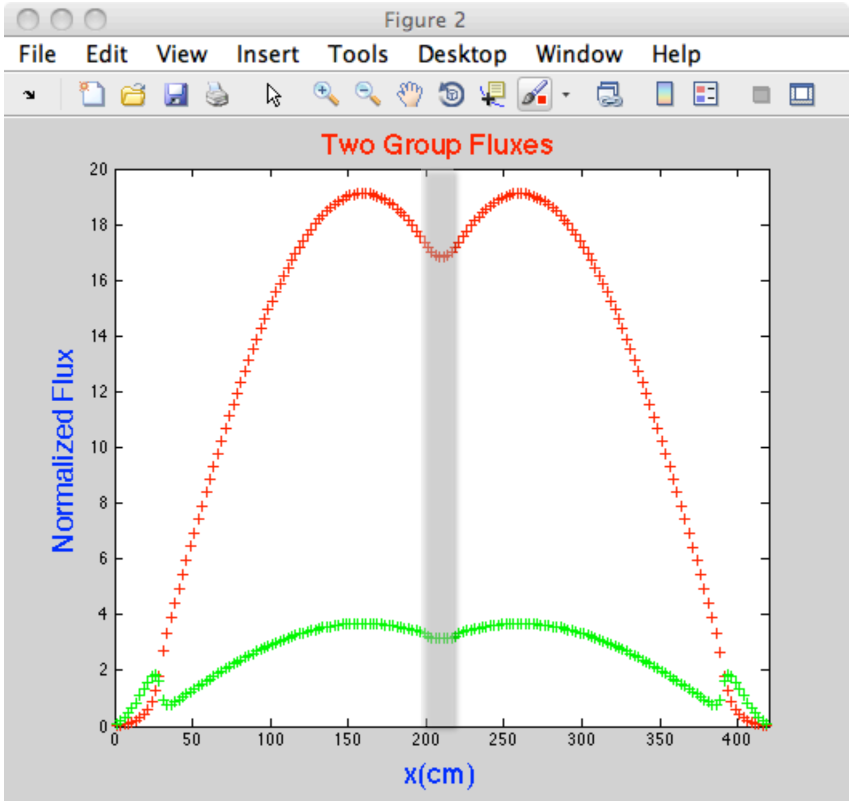
PSET # 3
SOLVING 2-GROUP TRANSIENT FINITE-DIFFERENCE EQUATIONS

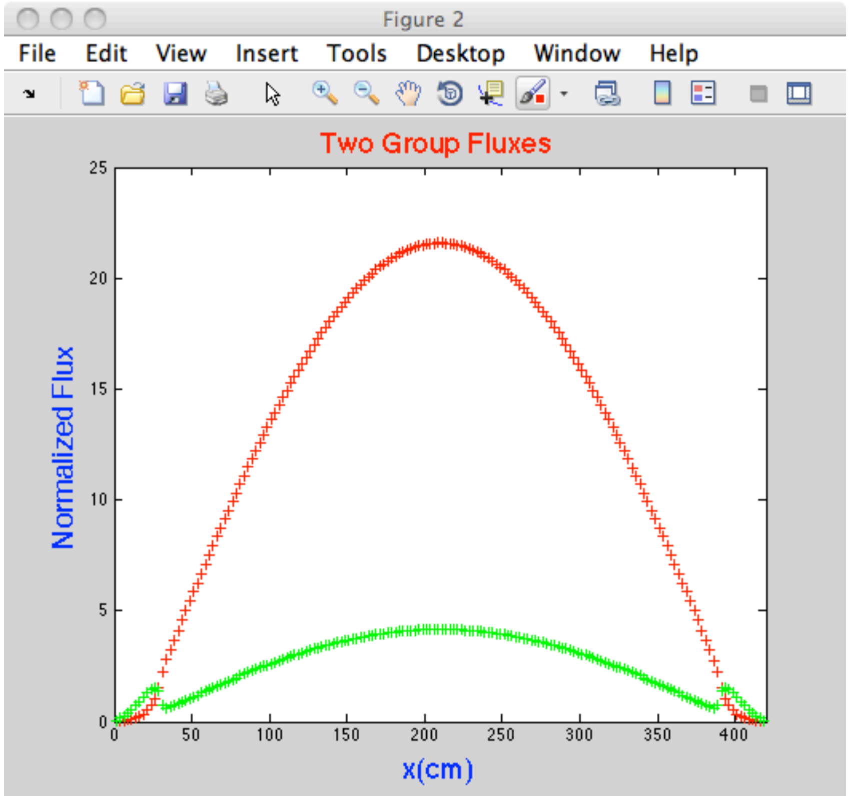
Due: Oct. 1, 2012

Massachusetts Institute of Technology

Pset 3: Reference Transient Solutions for Time Integration Testing

Transient Control Rod Withdrawal in a 1-D Reflected Core





Dynamic Simulations in 1-D
No Feedback Cases for:
Delayed Critical
Prompt Critical

Pset 3

Solve 1-D, 2-Group Diffusion Problem for Rod Withdrawal and Insertion for the Following Materials, Geometry, and Perturbations

D1, D2, Sigma-a1, Sigma-a2, Sig1→2, Nu*sigma-f1, Nu*sigma-f2

```
%
% define two group cross sections for all potential materials
%
nmat=6;
xs(1,1)=1.300; xs(2,1)=0.500; xs(3,1)=0.0098; xs(4,1)=0.114; xs(5,1)=0.022; xs(6,1)=0.006; xs(7,1)=0.1950;  % 3% enriched fuel
xs(1,2)=1.300; xs(2,2)=0.500; xs(3,2)=0.0105; xs(4,2)=0.134; xs(5,2)=0.022; xs(6,2)=0.008; xs(7,2)=0.2380;  % 4% enriched fuel
xs(1,3)=1.500; xs(2,3)=0.500; xs(3,3)=0.0002; xs(4,3)=0.010; xs(5,3)=0.032; xs(6,3)=0.000; xs(7,3)=0.0000;  % water
xs(1,4)=1.300; xs(2,4)=0.500; xs(3,4)=999.99; xs(4,4)=999.9; xs(5,4)=0.020; xs(6,4)=0.000; xs(7,4)=0.0000;  % black absorber
xs(1,5)=1.300; xs(2,5)=0.500; xs(3,5)=0.0098; xs(4,5)=0.118; xs(5,5)=0.022; xs(6,5)=0.006; xs(7,5)=0.1950;  % 3% enriched + rod1
xs(1,6)=1.300; xs(2,6)=0.500; xs(3,6)=0.0150; xs(4,6)=0.145; xs(5,6)=0.022; xs(6,6)=0.006; xs(7,6)=0.1950;  % 3% enriched + rod2
%
% define problem geometry and assign materials:
%
%   #zones=NZONE w(nzone),n(nzone), mat(nzone)
%
%  bc | slab 1| slab 2| slab 3| ..... slab(NZONE) | bc
%
NZONE=5;                                                % numer of material zones
w(1)= 30; w(2)=170; w(3)=20;  w(4)=170; w(5)=30;        % width    per zone
n(1)= 3;  n(2)=17;  n(3)=2;   n(4)=17;  n(5)=3;         % mesh     per zone
m(1)= 3;  m(2)=1 ;  m(3)=5;   m(4)=1;   m(5)=3;         % material per zone problem 1
m(1)= 3;  m(2)=1 ;  m(3)=6;   m(4)=1;   m(5)=3;         % material per zone problem 2

n=n*10;                                                  % mesh refinement factor
bc=2;                                                   % 0=zero flux, 1=zero incoming, 2=reflective bc
```

Rod for problem #1
Rod for problem #2

1. b.c. | Reflector | Fuel-1 | Rod | Fuel-1| Reflector | b.c.
2. Cross sections, zone widths, and cross section given above
3. Reflective b.c. on outer surfaces (same as steady-state in Pset 2)

Pset 3
Fix the velocity error from lecture 4

```
%
% kinetics parameters
%
betai   = [.000218 .001023 .000605 .00131 .00220 .00060 .000540 .000152];   % no units, 8-group data
halfli = [55.6       24.5     16.3     5.21    2.37   1.04    0.424    0.195 ];   % half-life in seconds
decayi = log(2)./halfli;                                                  % sec-1
vel1=2200.*100.*(.100e4/.0253)^.5; % cm/sec
vel2=2200.*100.*(.100  /.0253)^.5; % cm/sec
beta= sum(betai);  fprintf ('\n%s',' beta-effective - '); fprintf ('%14.7f',beta);

    if ido==5
```

Pset 3 (Part A)

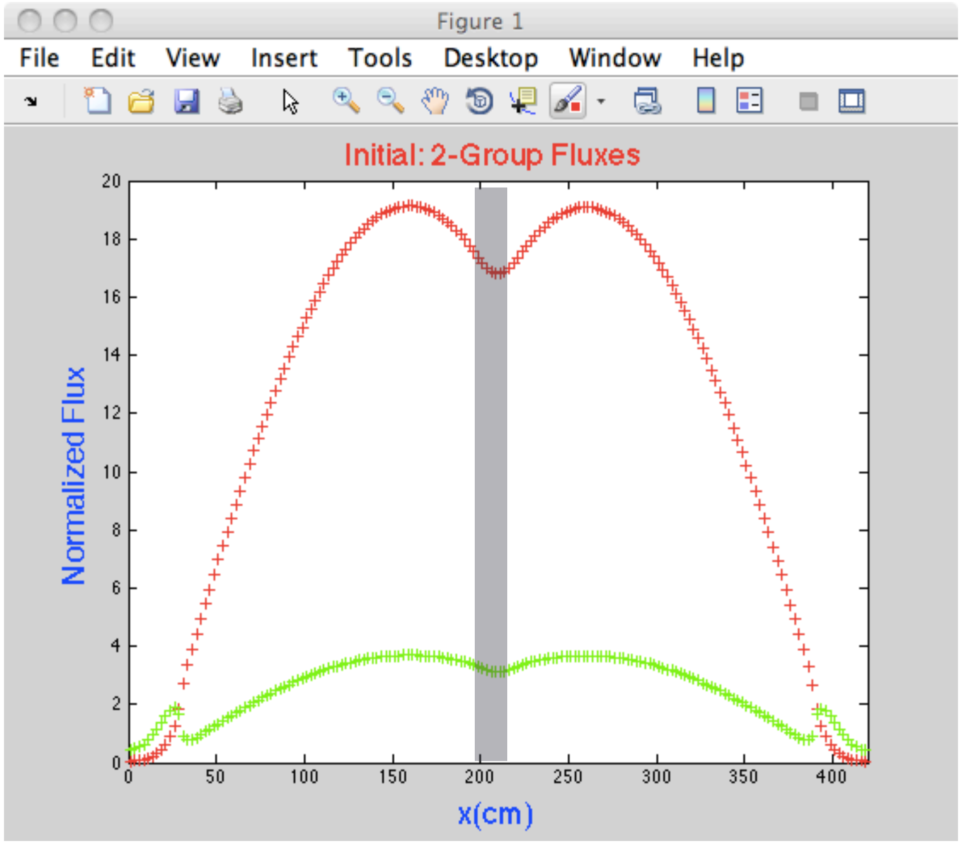
Write your own transient diffusion solver (in any language you choose) using MATLAB inversion, P-J, or G-S iterative flux inversion

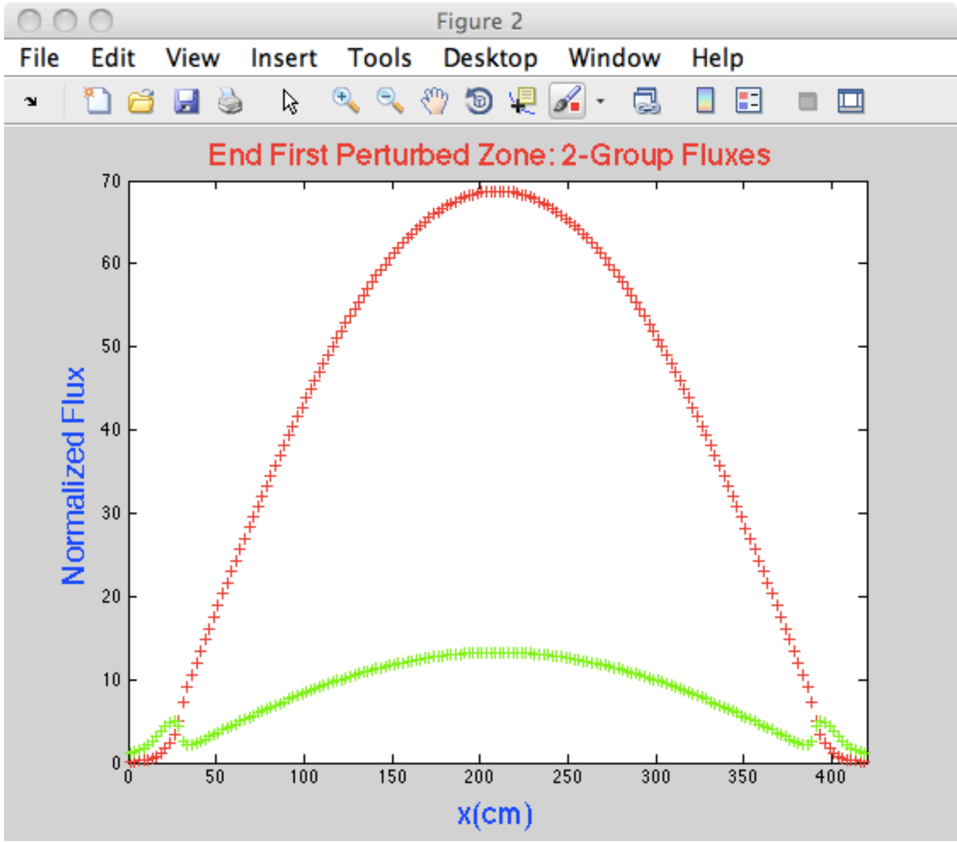
PART A: Delayed Critical Bank Withdrawal

1. Assume solution is spatially converged with 1.0 cm spatial mesh.
2. Withdraw the control rod uniformly over t = [2.00,4.00] seconds
3. Insert the control rod uniformly over t = [10.0,12.0] seconds
4. Follow the transient until t=50.0 sec.

- What is the static rod worth (in fraction of beta)?
- What uniform time step size is required to converge the peak core power to 1%?
- What uniform time step size is required to converge the final core power to 1%?
- Plot normalized core power vs. time for the converged time-step (both powers).
- Plot fractional error in peak and final core powers vs. time step when the converged time step is multiplied by 1, 2, 4, 8, 16, 32, and 64.
- Plot fractional error in peak and final core powers vs. spatial mesh (using converged time step) for mesh of 1.0, 2.0, 5.0, and 10.0 cm.

Pset 3 (Part A)





Delayed Critical Rod Bank Withdrawal

Pset 3 (Part A)

Evaluate

Pset 3 (Part A)

Write your own transient diffusion solver (in any language you choose) using MATLAB inversion, P-J, or G-S iterative flux inversion

PART A: Delayed Critical Bank Withdrawal

1. Assume solution is spatially converged with 1.0 cm spatial mesh.
2. Withdraw the control rod uniformly over t = [2.00,4.00] seconds
3. Insert the control rod uniformly over t = [10.0,12.0] seconds
4. Follow the transient until t=50.0 sec.

- What is the static rod worth (in fraction of beta)?
- What uniform time step size is required to converge the peak core power to 1%?
- What uniform time step size is required to converge the final core power to 1%?
- Plot normalized core power vs. time for the converged time-step (both powers).
- Plot fractional error in peak and final core powers vs. time step when the converged time step is multiplied by 1, 2, 4, 8, 16, 32, and 64.
- Plot fractional error in peak and final core powers vs. spatial mesh (using converged time step) for mesh of 1.0, 2.0, 5.0, and 10.0 cm.

Pset 3 (Part B)

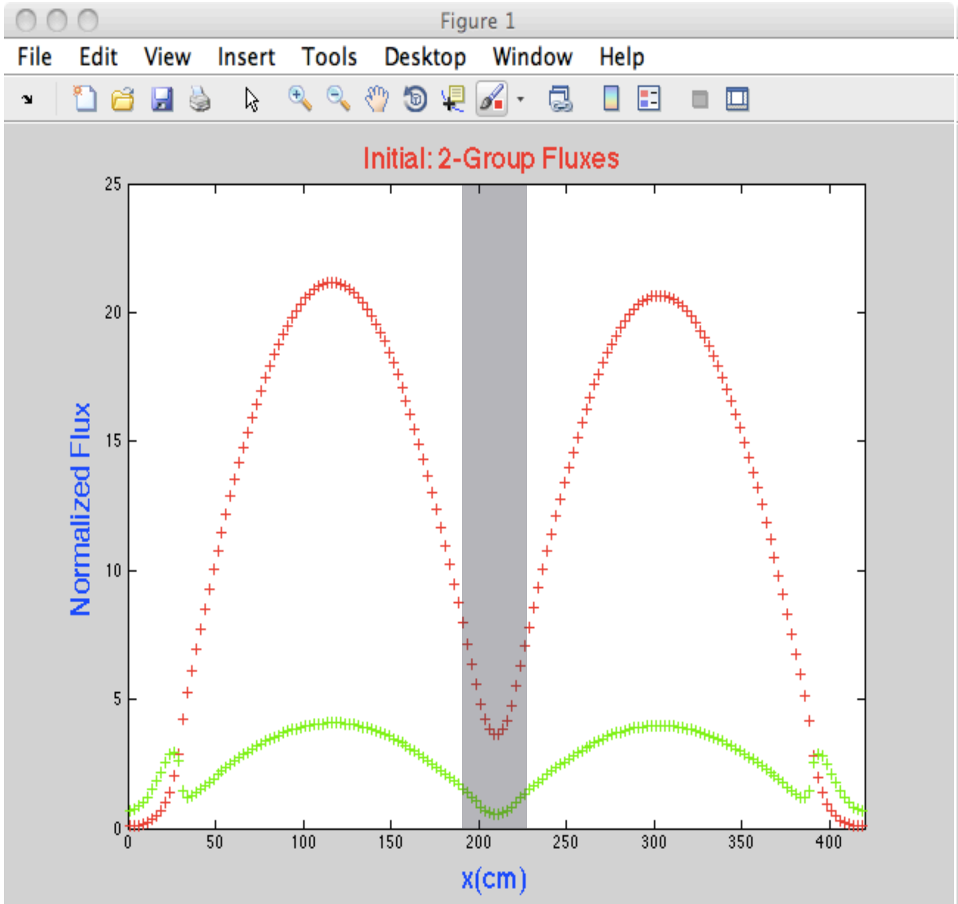
Write your own transient diffusion solver (in any language you choose) using MATLAB inversion, P-J, or G-S iterative flux inversion

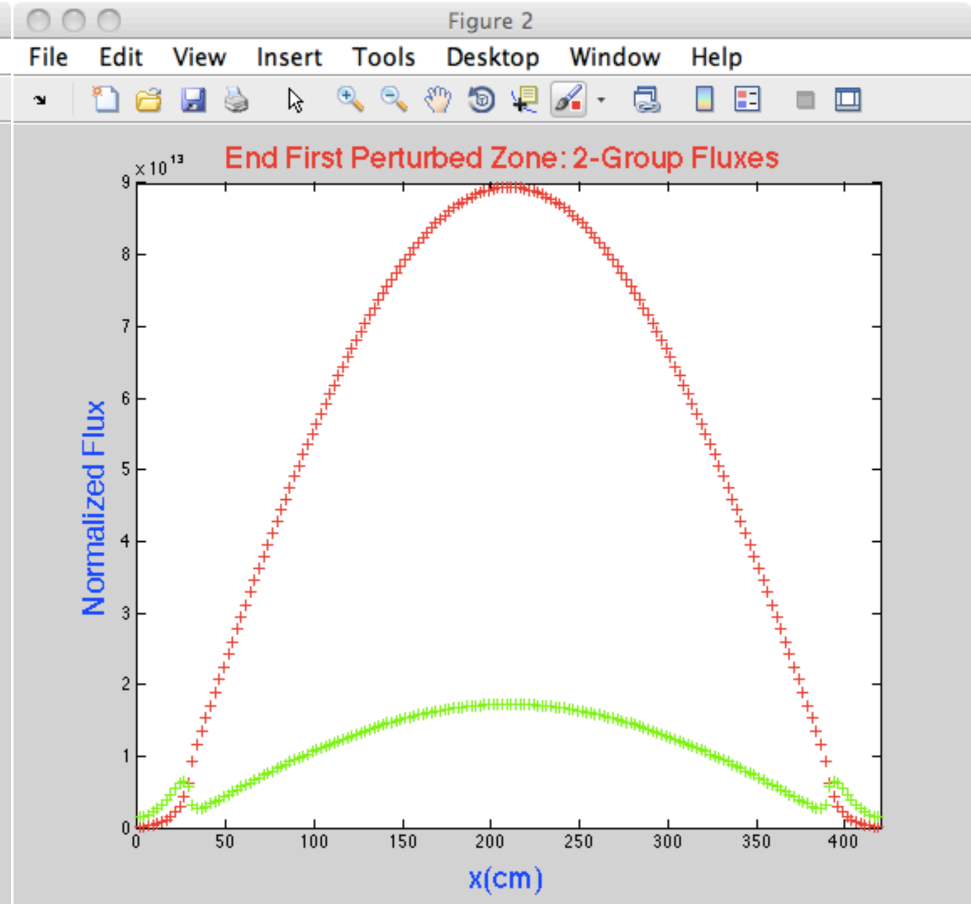
PART B: Super-Prompt Critical Rod Withdrawal and Scram

1. Assume solution is spatially converged with 1.0 cm spatial mesh.
2. Withdraw the control rod uniformly over t = [0.10,0.20] seconds
3. Scram the control rod uniformly over t = [0.50,1.50] seconds
4. Follow the transient until t=2.0 sec.

- What is the static rod worth (in fraction of beta)?
- What is dominance ratio of the static rod-inserted base case?
- What uniform time step size is required to converge the peak core power to 10%?
- What uniform time step size is required to converge the final core power to 1%?
- Plot normalized core power vs. time for the converged time-step (both powers).
- Plot fractional error in peak and final core powers vs. time step when the converged time step is multiplied by 1, 2, 4, 8, 16, 32, and 64.
- Plot fractional error in peak and final core powers vs. spatial mesh (using converged time step) for mesh of 1.0, 2.0, 5.0, and 10.0 cm.

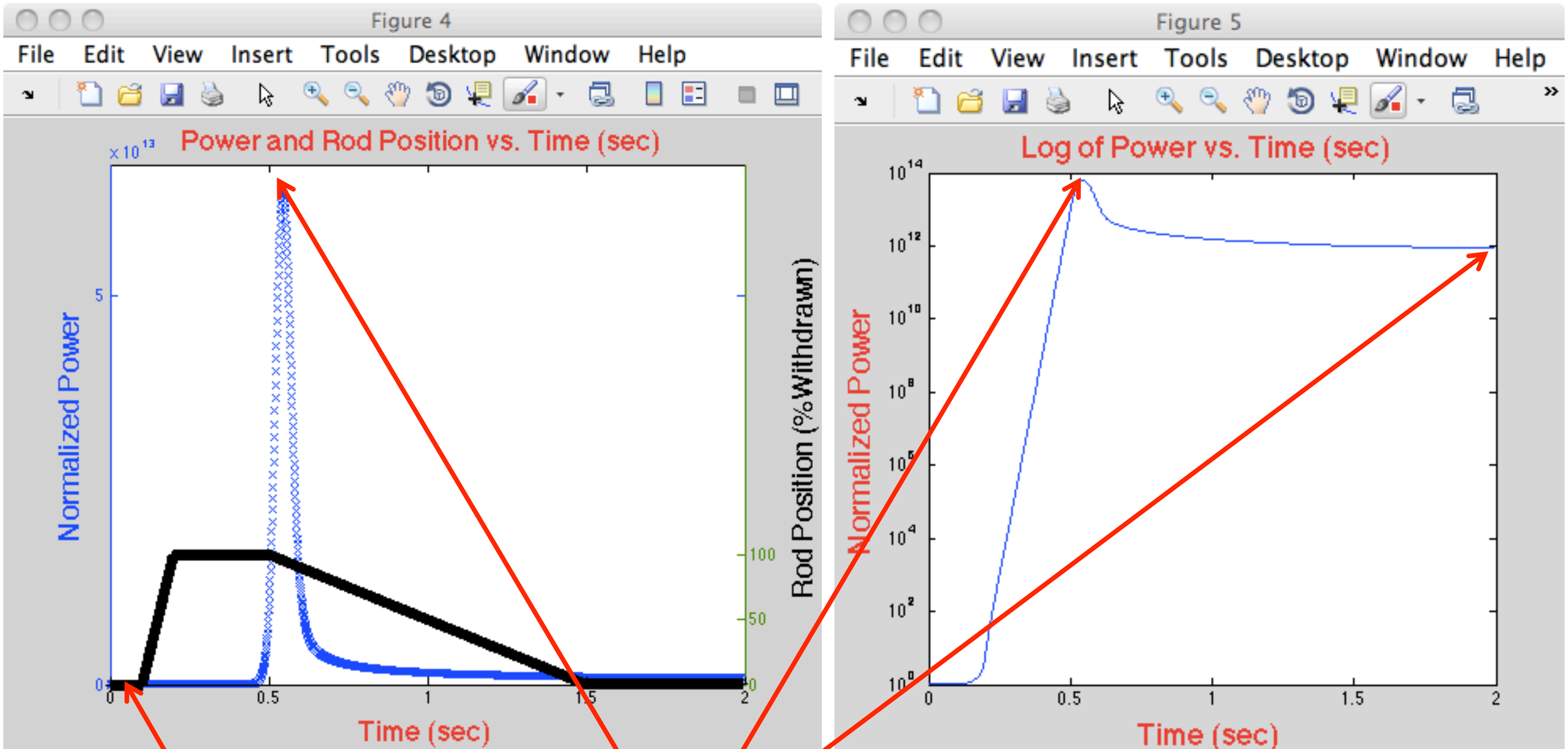
Pset 3 (Part B)





Super Prompt Critical Rod Bank Withdrawal

Pset 3 (Part B)



Make sure
to hold steady state

Evaluate

Pset 3 (Part B)

Write your own transient diffusion solver (in any language you choose) using MATLAB inversion, P-J, or G-S iterative flux inversion

PART B: Super-Prompt Critical Rod Withdrawal and Scram

1. Assume solution is spatially converged with 1.0 cm spatial mesh.
2. Withdraw the control rod uniformly over t = [0.10,0.20] seconds
3. Scram the control rod uniformly over t = [0.50,1.50] seconds
4. Follow the transient until t=2.0 sec.

- What is the static rod worth (in fraction of beta)?
- What is the dominance ratio of the static rod-inserted base case?
- What uniform time step size is required to converge the peak core power to 10%?
- What uniform time step size is required to converge the final core power to 1%?
- Plot normalized core power vs. time for the converged time-step (both powers).
- Plot fractional error in peak and final core powers vs. time step when the converged time step is multiplied by 1, 2, 4, 8, 16, 32, and 64.
- Plot fractional error in peak and final core powers vs. spatial mesh (using converged time step) for mesh of 1.0, 2.0, 5.0, and 10.0 cm.

Transient Finite-Difference 2-group Neutron Diffusion Equations

If we redefine "pseudo fission" and "pseudo removal" cross sections

$$\upsilon\hat{\Sigma}_{f,g}(\vec{r}) \equiv \left(\frac{[1-\beta(\vec{r})]}{k_{crit}} + \sum_{i}^{I} \frac{\beta_i(\vec{r})\lambda_i\Delta_t}{(1+\lambda_i\Delta_t)k_{crit}} \right) \upsilon\Sigma_{f,g}(\vec{r})$$

$$\hat{\Sigma}_{r,1}(\vec{r}) \equiv \left\langle \Sigma_{r,1}(\vec{r}) + \frac{1}{v_1(\vec{r})\Delta_t} \right\rangle \qquad \hat{\Sigma}_{r,2}(\vec{r}) \equiv \left\langle \Sigma_{a,2}(\vec{r}) + \frac{1}{v_2(\vec{r})\Delta_t} \right\rangle$$

$$S_1 \equiv \left\langle \frac{\phi_1(\vec{r},t_n)}{v_1(\vec{r})\Delta_t} + \sum_{i}^{I} \lambda_i \frac{C_i(\vec{r},t_n)}{(1+\lambda_i\Delta_t)} \right\rangle \qquad S_2 \equiv \left\langle \frac{\phi_2(\vec{r},t_n)}{v_2(\vec{r})\Delta_t} \right\rangle$$

And collect some terms:

$$-\nabla \cdot D_1(\vec{r})\nabla\phi_1(\vec{r},t_{n+1}) + \left(\hat{\Sigma}_{r,1}(\vec{r}) - \upsilon\hat{\Sigma}_{f,1}(\vec{r})\right)\phi_1(\vec{r},t_{n+1}) - \upsilon\hat{\Sigma}_{f,2}(\vec{r})\,\phi_2(\vec{r},t_{n+1}) = S_1$$

$$-\nabla \cdot D_2(\vec{r})\nabla\phi_2(\vec{r},t_{n+1}) + \hat{\Sigma}_{r,2}(\vec{r})\phi_2(\vec{r},t_{n+1}) - \Sigma_{1\to 2}(\vec{r})\,\phi_1(\vec{r},t_{n+1}) = S_2$$

$$C_i(\vec{r},t_{n+1}) = \frac{\beta_i(\vec{r})\Delta_t}{(1+\lambda_i\Delta_t)k_{crit}} \left[\upsilon\Sigma_{f,1}(\vec{r})\,\phi_1(\vec{r},t_{n+1}) + \upsilon\Sigma_{f,2}(\vec{r})\,\phi_2(\vec{r},t_{n+1}) \right] + \frac{C_i(\vec{r},t_n)}{(1+\lambda_i\Delta_t)}, \qquad i = 1, \ldots, I$$

Transient vs. Steady State Finite-Difference 2-group Matrix Equations

1	2	3	4	5	6	7	8	...	...	...	N

$$-\hat{D}_1^{m-1.m}\phi_1^{m-1}(t_{n+1}) + \left[\left(\hat{\Sigma}_{r1}^m - \upsilon\hat{\Sigma}_{f1}^m\right)\Delta_x + \hat{D}_1^{m-1.m} + \hat{D}_1^{m,m+1}\right]\phi_1^m(t_{n+1}) - \hat{D}_1^{m,m+1}\phi_1^{m+1}(t_{n+1}) - \upsilon\hat{\Sigma}_{f2}^m\Delta_x\phi_2^m(t_{n+1}) = S_1^m(t_n)\Delta_x$$

$$-\hat{D}_2^{m-1.m}\phi_2^{m-1}(t_{n+1}) + \left[\hat{\Sigma}_{r2}^m\Delta_x + \hat{D}_2^{m-1.m} + \hat{D}_2^{m,m+1}\right]\phi_2^m(t_{n+1}) - \hat{D}_2^{m,m+1}\phi_2^{m+1}(t_{n+1}) - \Sigma_{12}^m\Delta_x\phi_1^m(t_{n+1}) = S_2^m(t_n)\Delta_x$$

Each group flux matrix equation is tri-diagonal (ignoring group transfer terms):

$$\begin{bmatrix} \left[L+\hat{D}+U\right]_1 & -[\hat{M}]_2 \\ -[T]_2 & \left[L+\hat{D}+U\right]_1 \end{bmatrix} \begin{bmatrix} [\phi_1] \\ [\phi_2] \end{bmatrix}^{t_{n+1}} = \begin{bmatrix} [S_1] \\ [S_2] \end{bmatrix}^{t_n}$$

$$C_i^m(t_{n+1}) = \frac{\beta_i^m(\vec{r})\Delta_t}{(1+\lambda_i\Delta_t)k_{crit}}\left[\upsilon\Sigma_{f1}^m\phi_1^m(t_{n+1}) + \upsilon\Sigma_{f2}^m\phi_2^m(t_{n+1})\right] + \frac{C_i^m(t_n)}{(1+\lambda_i\Delta_t)}, \qquad i = 1,\dots,I$$

Each precursor matrix equation is strictly diagonal (no spatial coupling):

$$\left[C_i\right] = \left[D_i\right]\left\{[M]_1\left[\phi_1\right]^{t_{n+1}} + [M]_2\left[\phi_1\right]^{t_{n+1}}\right\} + \left[Q_i\right]^{t_n}, \qquad i = 1,\dots,I$$

MATLAB coding of transient case: Setup Transient Matrix and Solve

```
%
% solve transient problem
%
deltat=0.1  ;  tend=50.0 ; nstep=tend/deltat;                          % set time integration parameters
pertsl=2.0  ; pertel=4.0 ; pertxsl=+1.00*(xs(4,1)-xs(4,5))/(pertel-pertsl);  % set perturbationl start, end del-xsa2
power([1:nstep])=0; time([1:nstep])=0; C([1:8,1:2*NP])=0; S([1:2*NP])=0;

M0=M/keff; FS=(M0*phi); power0=sum(FS);                     % save real fission matrix
for i=1:8
    C(i,:)=(FS')*betai(i)/decayi(i);                        % initialize precursors [volume in M0, so in C(i)]
end
for n=1:NP;
   A(n,n)        =A(n,n)       +h(n)/(vel1*deltat);         % add pseudo absorption to group 1 removal
   A(NP+n,NP+n)=A(NP+n,NP+n)+h(n)/(vel2*deltat);            % add pseudo absorption to group 2 removal
end
factor=(1.-beta)/keff; for i=1:8; factor=factor+betai(i)*decayi(i)*deltat/((1.+decayi(i)*deltat)*keff); end
M=M*factor;                                                 % create pseduo fission matrix
AA=A-M;                                                     % create base iteration coefficient matrix
for step=1:nstep; time(step)=deltat*step;                   % loop over time steps
   for n=1:NP
     if (matvec(n) == 3)
       if (time(step) > pertsl && time(step)<= pertel)
         AA(NP+n,NP+n)=AA(NP+n,NP+n)+h(n)*pertxsl*deltat;   % move control rod out
       end
     end
     S(NP+n)=h(n)*phi(NP+n)/(vel2*deltat);                  % compute group 2 source vector
     S(n   )= h(n)*phi(n   )/(vel1*deltat);                 % compute group 1 source vector
     for i=1:8;
         S(n)=S(n)+decayi(i)*C(i,n)/(1.+decayi(i)*deltat);
     end
   end
   phi=(AA^(-1))*S';                                        % evaluate new time step flux vector
   FS=(M0*phi); power(step)=sum(FS);                        % compute normalize core poweer
   for i=1:8
      C(i,:)=(C(i,:)+(FS')*betai(i)*deltat)/(1.+decayi(i)*deltat);   % new time step precursor vector
   end
end
```

$$\upsilon\hat{\Sigma}_{f,g}(\vec{r}) \equiv \left(\frac{[1-\beta(\vec{r})]}{k_{crit}} + \sum_i^I \frac{\beta_i(\vec{r})\lambda_i\Delta_t}{(1+\lambda_i\Delta_t)k_{crit}} \right) \upsilon\Sigma_{f,g}(\vec{r})$$

$$\hat{\Sigma}_{r,1}(\vec{r}) \equiv \left\langle \Sigma_{r,1}(\vec{r}) + \frac{1}{v_1(\vec{r})\Delta_t} \right\rangle \qquad \hat{\Sigma}_{r,2}(\vec{r}) \equiv \left\langle \Sigma_{a,2}(\vec{r}) + \frac{1}{v_2(\vec{r})\Delta_t} \right\rangle$$

$$S_1 \equiv \left\langle \frac{\phi_1(\vec{r},t_n)}{v_1(\vec{r})\Delta_t} + \sum_i^I \lambda_i \frac{C_i(\vec{r},t_n)}{(1+\lambda_i\Delta_t)} \right\rangle \qquad S_2 \equiv \left\langle \frac{\phi_2(\vec{r},t_n)}{v_2(\vec{r})\Delta_t} \right\rangle$$

$$\begin{bmatrix} \left[L+\hat{D}+U\right]_1 & -[\hat{M}]_2 \\ -[T]_2 & \left[L+\hat{D}+U\right]_1 \end{bmatrix} \begin{bmatrix} [\phi_1] \\ [\phi_2] \end{bmatrix}^{t_{n+1}} = \begin{bmatrix} [S_1] \\ [S_2] \end{bmatrix}^{t_n}$$

$$\left[C_i\right] = \left[D_i\right]\left\{ [M]_1 \left[\phi_1\right]^{t_{n+1}} + [M]_2 \left[\phi_1\right]^{t_{n+1}} \right\} + \left[Q_i\right]^{t_n}, \qquad i = 1, \dots, I$$

Let's Examine Point Kinetics Solutions

- Compare true time-dependent solution to:
  - PKE solution using unity-weighted initial steady-state fluxes
  - PKE solution using unity-weighted final steady-state fluxes
  - PKE solution using adjoint-weighted initial steady-state fluxes
  - PKE solution using adjoint-weighted final steady-state fluxes
  - 2-group PKE solution using unity-weighted initial fluxes
  - 2-group PKE solution using unity-weighted final fluxes
  - Geometrically-interpolated shape- and weight-function PKEs
  - Quasi-Static solution
  - Improved Quasi-Static solution

Numerical Solution of Inverse Kinetics Equations

- First, let's examine reactivity from IKE to see what spatial kinetics reactivity is:

$$[C(t)] = [e^{-At}][C_0] + [e^{-At}][A]^{-1}\{[e^{At}][Y] - [Y_0]\}$$

$$where \quad Y(t) \equiv [\beta_i]\frac{T(t)}{\Lambda}$$

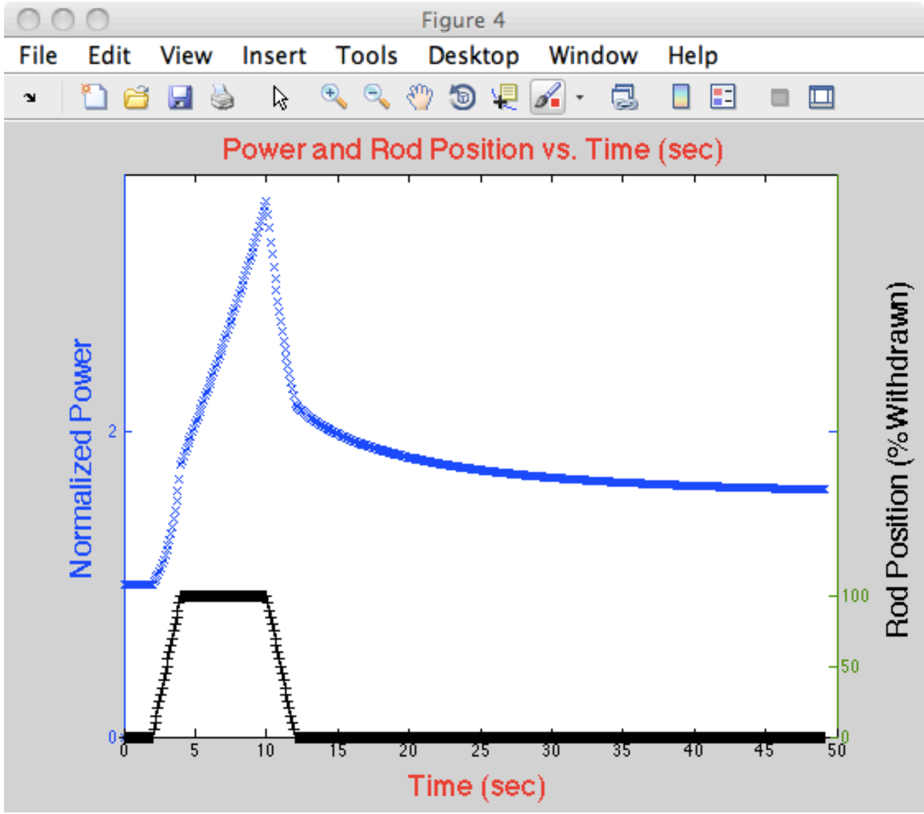
- Solve for time-varying concentrations of precursors for reactor power shape vs. time by applying this equation for discrete steps.

- From the PKEs, we solve for reactivity in terms of precursor concentration and reactor power vs. time by making finite-difference approximation

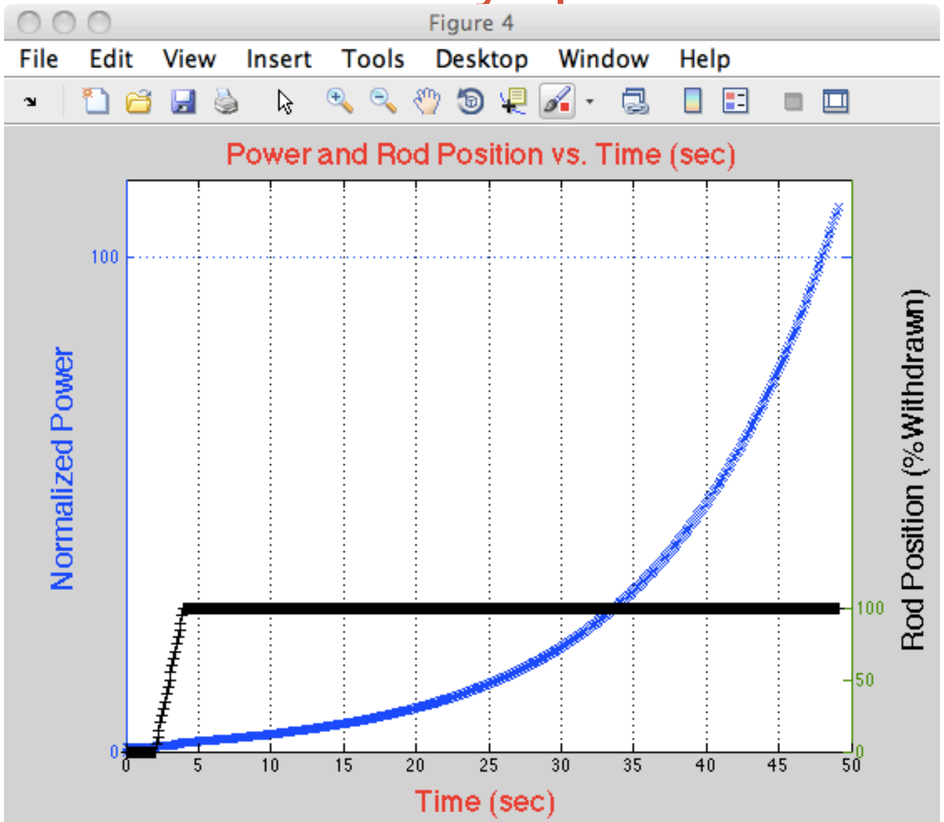
$$\frac{d}{dt}T(t) = \frac{\rho(t)-\beta}{\Lambda}T(t) + \sum_i \lambda_i C_i(t)$$

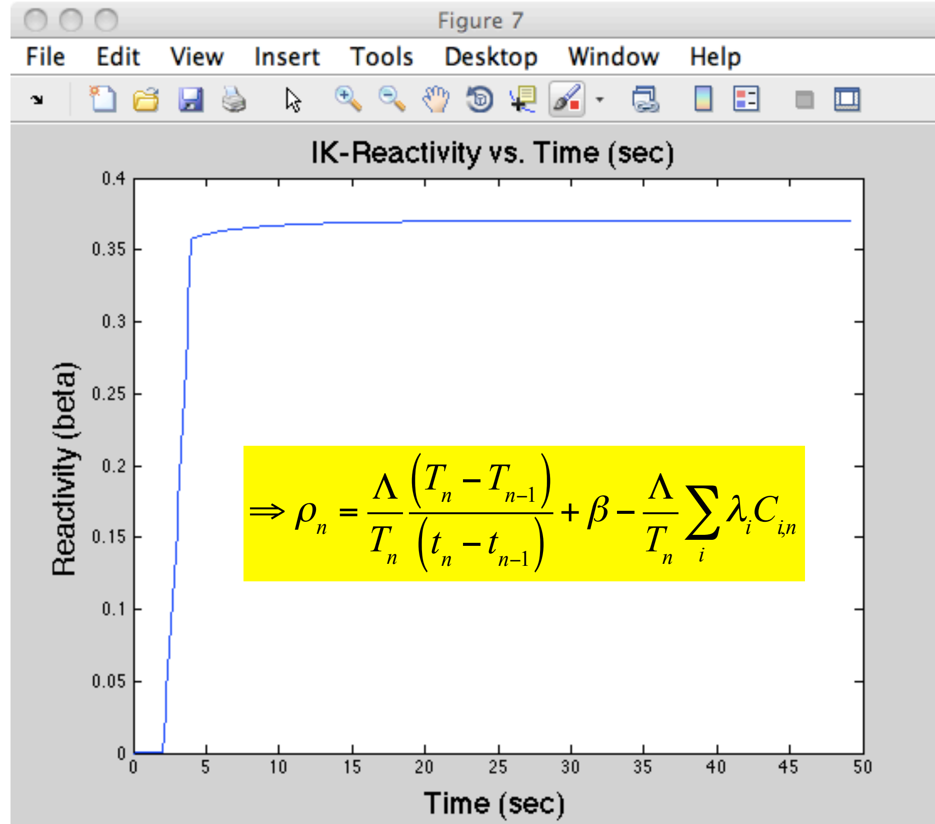
$$\Rightarrow \rho(t) = \frac{\Lambda}{T(t)}\frac{d}{dt}T(t) + \beta - \frac{\Lambda}{T(t)}\sum_i \lambda_i C_i(t)$$

$$\Rightarrow \rho_n = \frac{\Lambda}{T_n}\frac{(T_n - T_{n-1})}{(t_n - t_{n-1})} + \beta - \frac{\Lambda}{T_n}\sum_i \lambda_i C_{i,n}$$



Asymptotic Inverse Point Kinetics Solution





$$\rho_{IK} = 0.37230\beta$$

$$k_{eff}^{rodded} = 1.36460 \qquad k_{eff}^{unrodded} = 1.36799$$

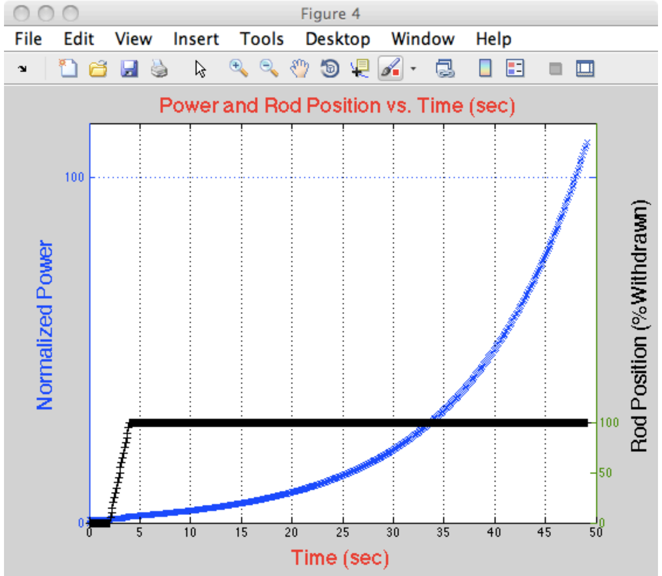
$$\rho_{static} = \frac{1.36799 - 1.36460}{1.36460\left(\beta\right)} = 0.37265\beta$$

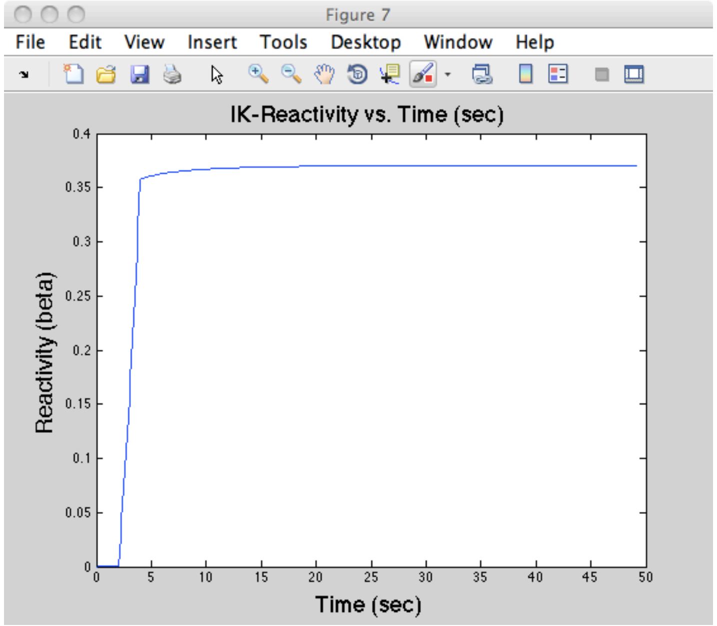
PKE Parameters Edited From Dynamic Fluxes

$$\rho(t) \equiv \frac{\left[\int d\vec{r}\ \phi_1^*(\vec{r},t)\left\langle -\hat{\Sigma}_{r,1}(\vec{r},t)\,\phi_1(\vec{r},t) + \left[\nu\Sigma_{f,1}(\vec{r},t)\,\phi_1(\vec{r},t) + \nu\Sigma_{f,2}(\vec{r},t)\,\phi_2(\vec{r},t)\right]\right\rangle + \int d\vec{r}\ \phi_2^*(\vec{r},t)\left\langle -\Sigma_{a,2}(\vec{r},t)\phi_2(\vec{r},t) + \hat{\Sigma}_{1\to 2}(\vec{r},t)\,\phi_1(\vec{r},t)\right\rangle\right]}{\int d\vec{r}\ \phi_1^*(\vec{r},t)\left[\nu\Sigma_{f,1}(\vec{r},t)\,\phi_1(\vec{r},t) + \nu\Sigma_{f,2}(\vec{r},t)\,\phi_2(\vec{r},t)\right]}$$

$$\beta_i(t) = \frac{\int d\vec{r}\ \phi_1^*(\vec{r})\beta_i(\vec{r},t)\left[\nu\Sigma_{f,1}(\vec{r},t)\,\phi_1(\vec{r},t) + \nu\Sigma_{f,2}(\vec{r},t)\,\phi_2(\vec{r},t)\right]}{\int d\vec{r}\ \phi_1^*(\vec{r},t)\left[\nu\Sigma_{f,1}(\vec{r},t)\,\phi_1(\vec{r},t) + \nu\Sigma_{f,2}(\vec{r},t)\,S_2(\vec{r},t)\right]\phi_2(\vec{r},t)} = \beta_i$$

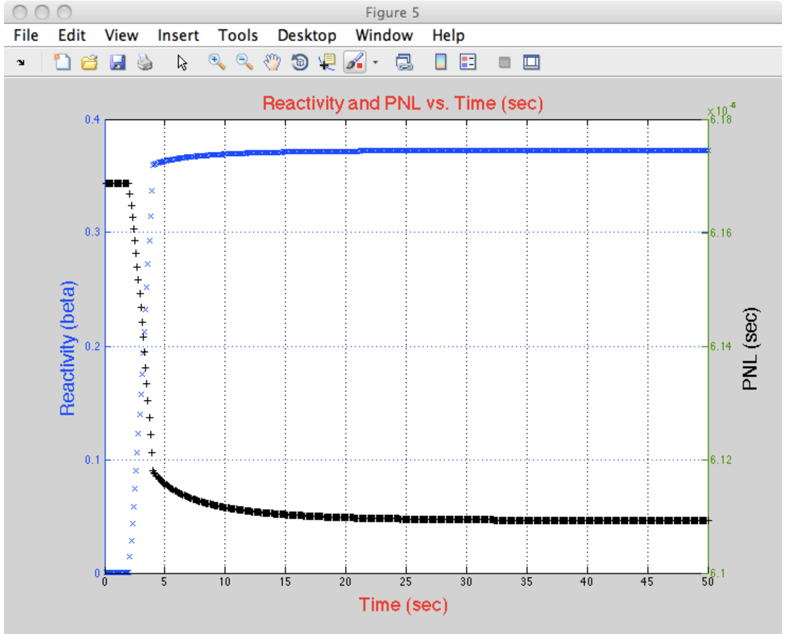
$$\Lambda(t) = \frac{\int d\vec{r}\ \phi_1^*(\vec{r},t)\left[\frac{1}{v_1}\phi_1(\vec{r},t) + \frac{1}{v_2}\phi_2(\vec{r},t)\right]}{\int d\vec{r}\ \phi_1^*(\vec{r},t)\left[\nu\Sigma_{f,1}(\vec{r},t)\,\phi_1(\vec{r},t) + \nu\Sigma_{f,2}(\vec{r},t)\,\phi_2(\vec{r},t)\right]}$$



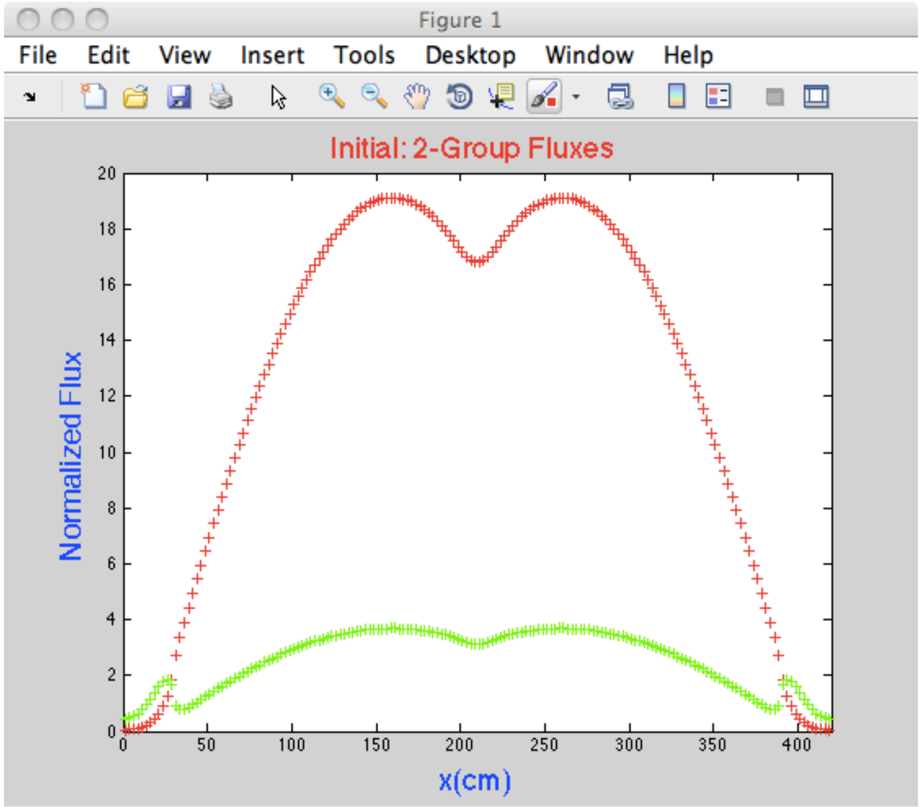


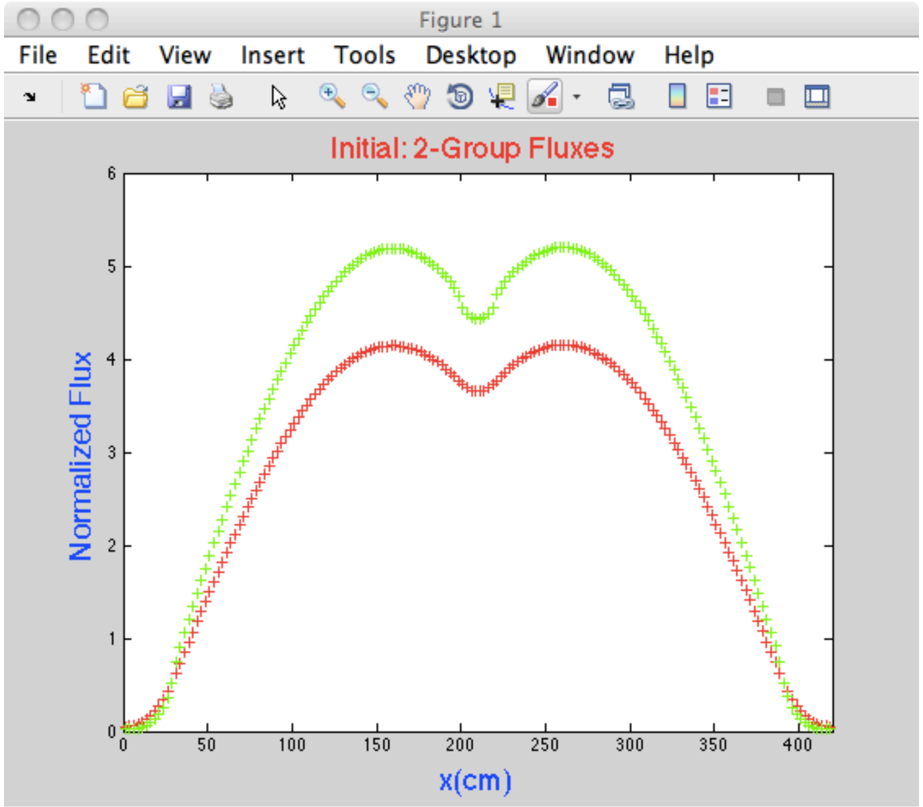
$$\rho_{static} = 0.37265\beta$$
$$\rho_{dynamic} = 0.37263\beta$$
$$\rho_{IK} = 0.37230\beta$$

Note: small time step sensitivity for IK's rho



PKE Parameters Dependence on Shape Function





Dynamic Fluxes

$$\rho_{static} = 0.37265\beta$$
$$\rho_{dynamic} = 0.37263\beta$$

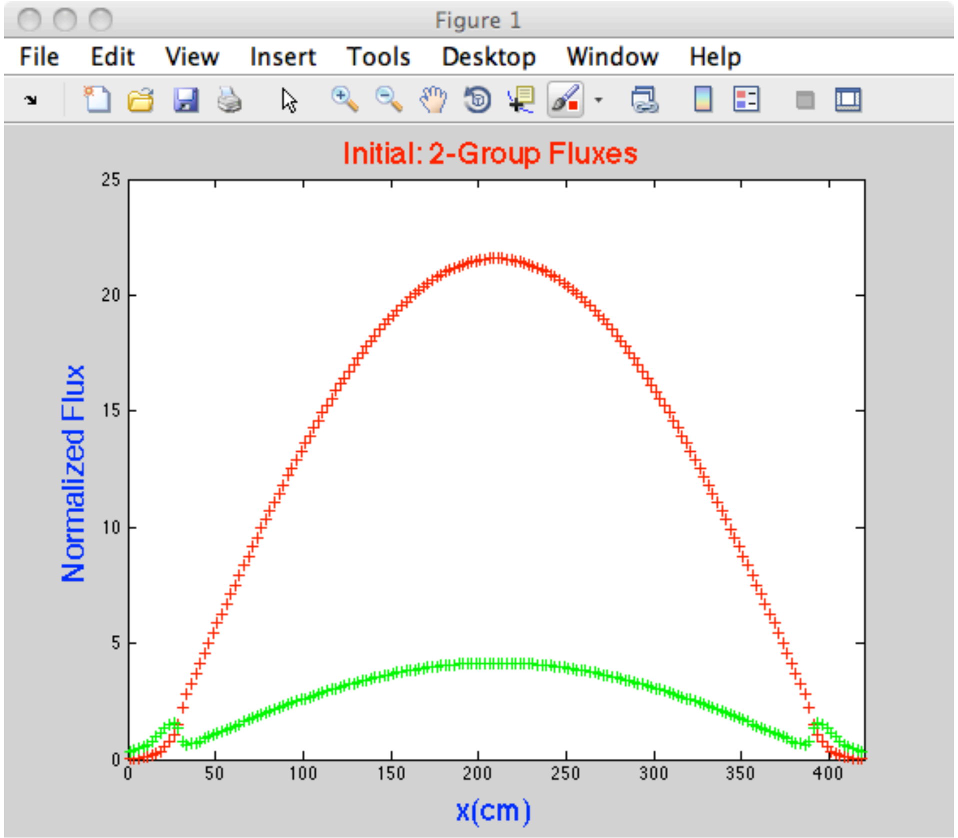
Rod-In Flux/Unity Adjoint

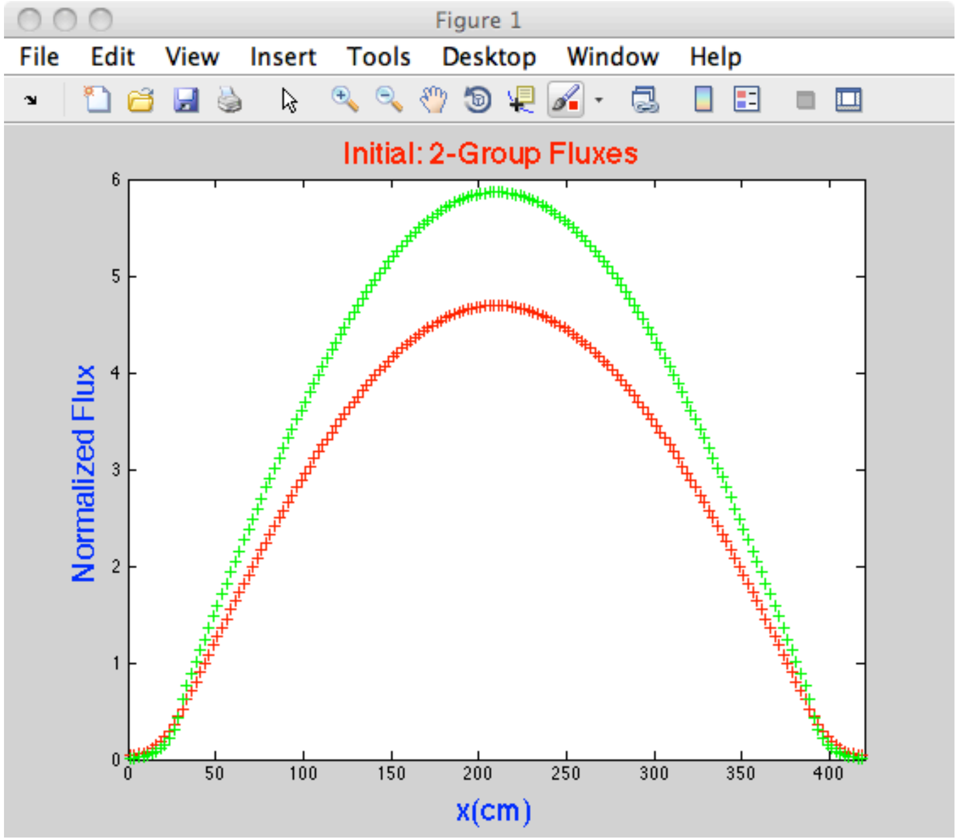
$$\rho_{static} = 0.37265\beta$$
$$\rho_{rod-in} = 0.22760\beta$$

Rod-in Flux and Adjoint

$$\rho_{static} = 0.37265\beta$$
$$\rho_{rod-in} = 0.29040\beta$$

PKE Parameters Dependence on Shape Function





Dynamic Fluxes

$\rho_{static} = 0.37265\beta$

$\rho_{dynamic} = 0.37263\beta$

Rod-out Flux/Unity Adjoint

$\rho_{static} = 0.37265\beta$

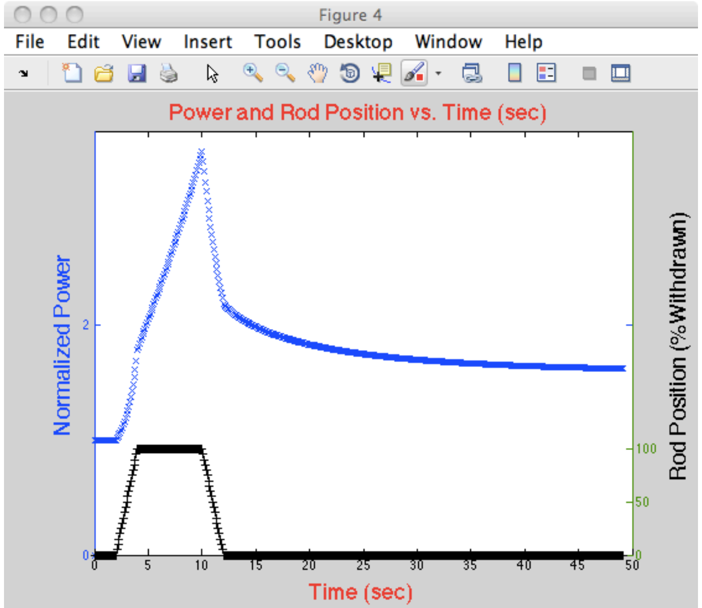
$\rho_{rod-out} = 0.37265\beta$

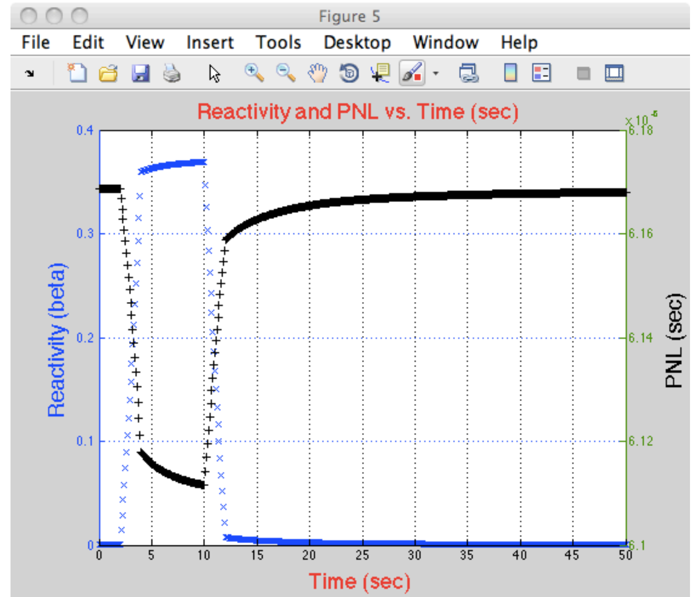
Rod-out Flux and Adjoint

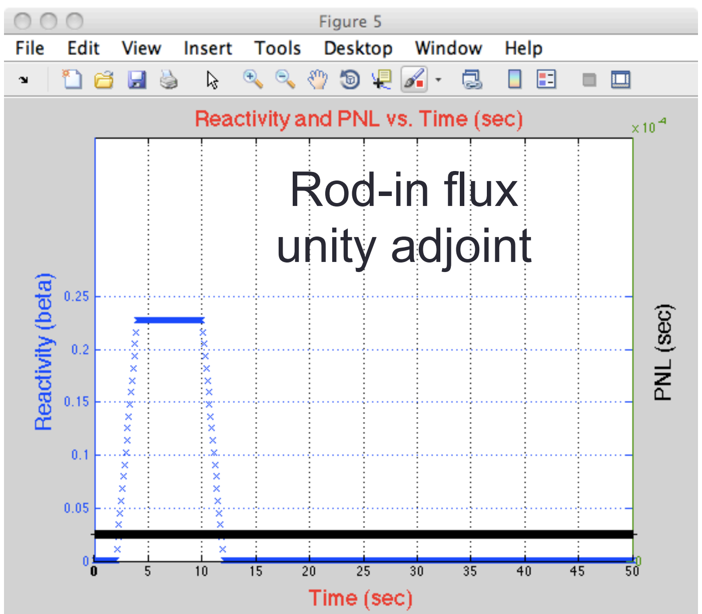
$\rho_{static} = 0.37265\beta$

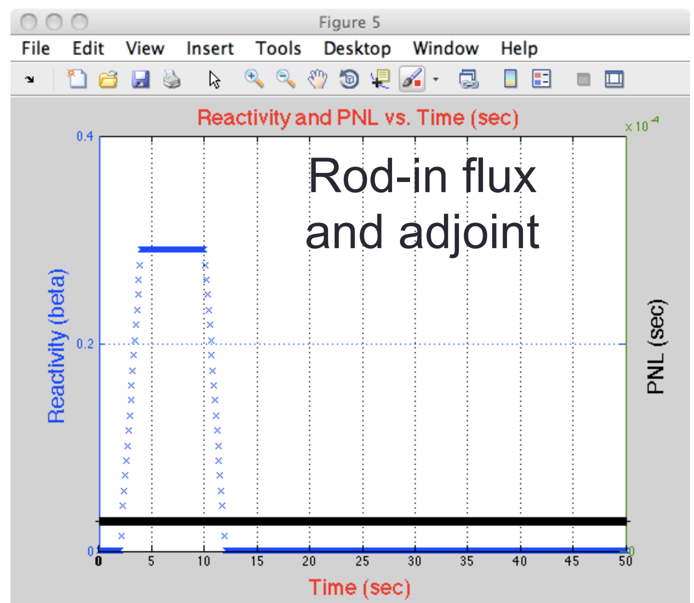
$\rho_{rod-out} = 0.37265\beta$

Rod in/out Transient Edits of Point Point Kinetics Reactivity vs. Time

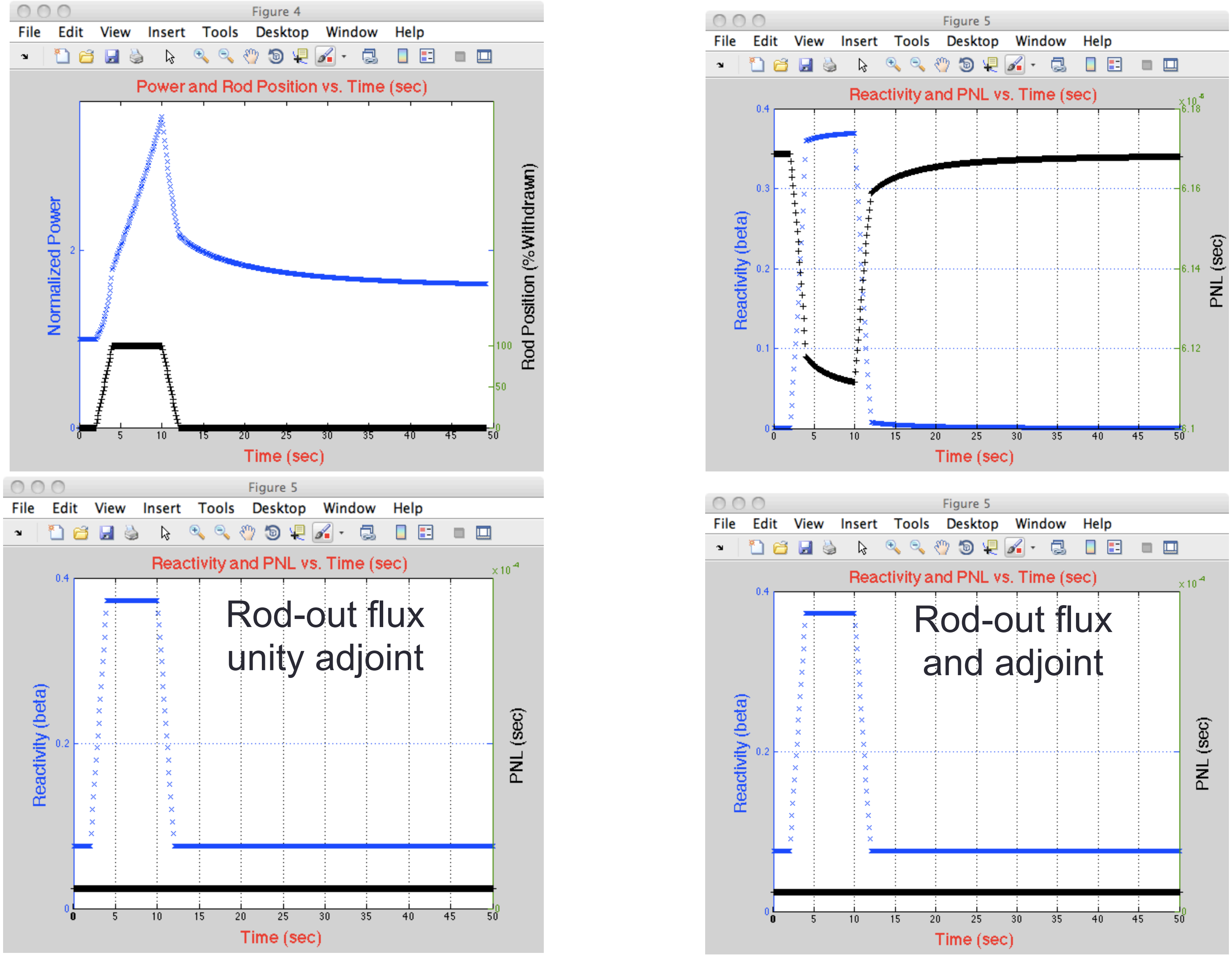




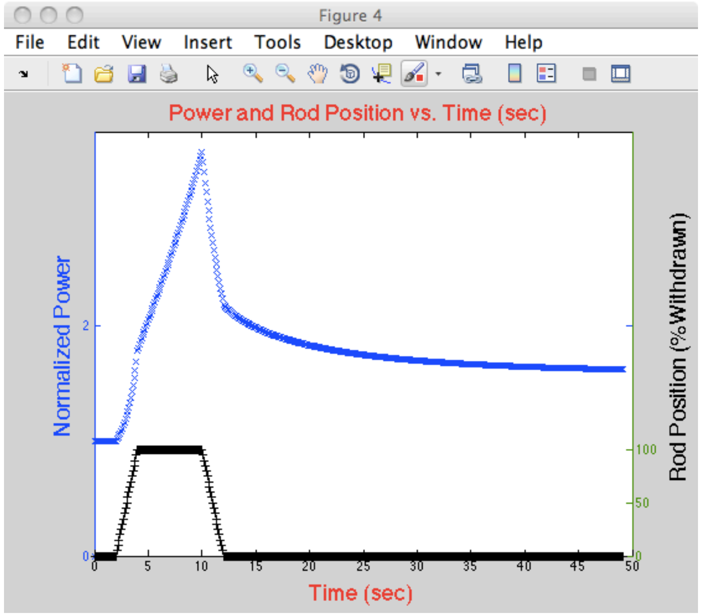


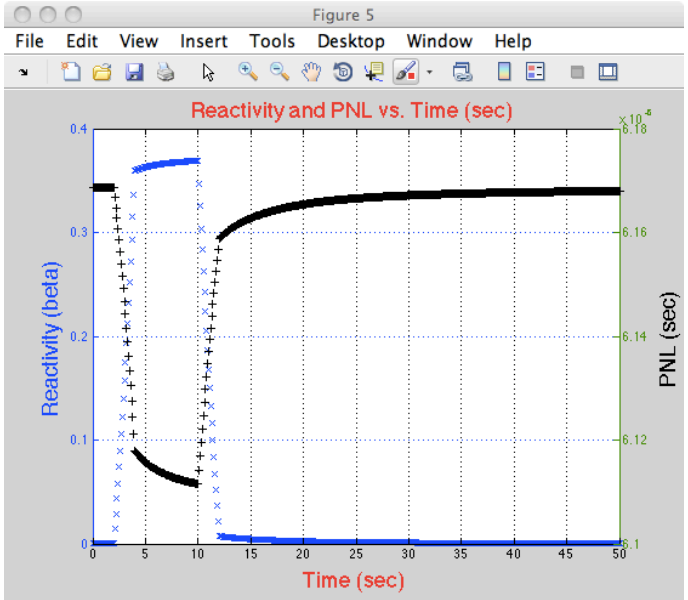


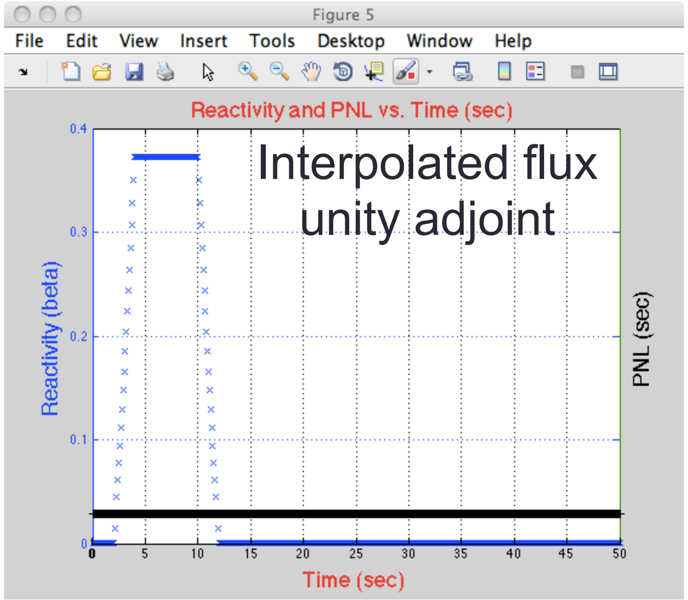
Rod in/out Transient Edits of Point Point Kinetics Reactivity vs. Time

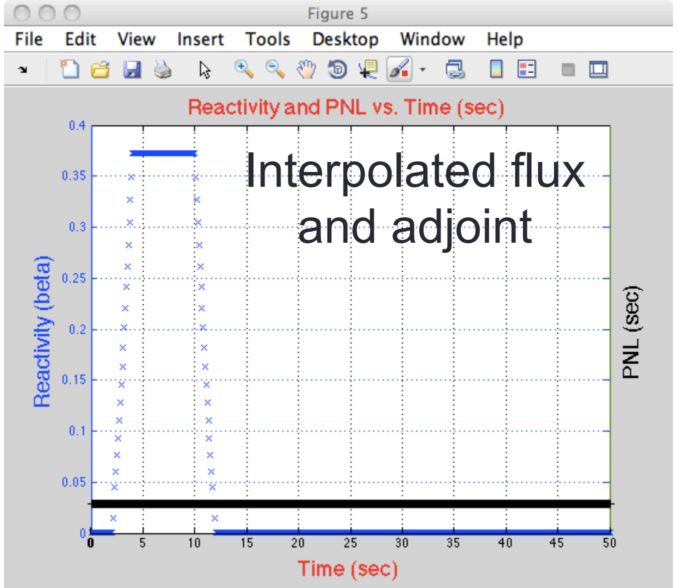


Geometrical-Interpolation Edits of Point Point Kinetics Reactivity vs. Time

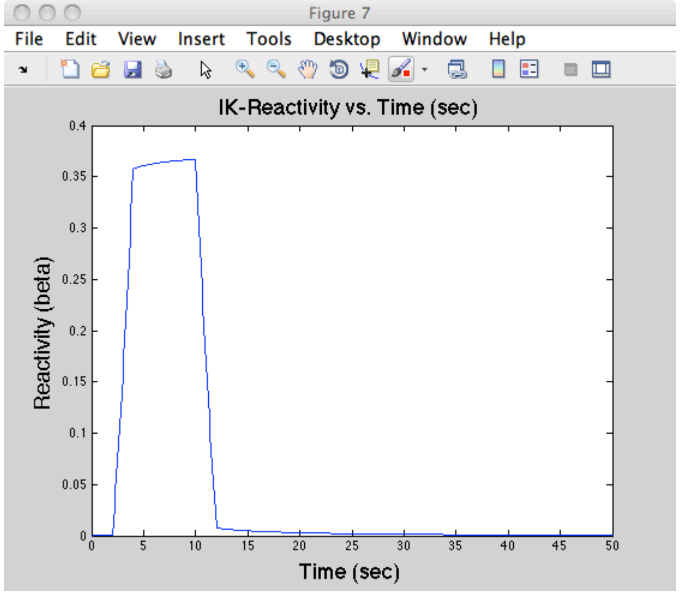


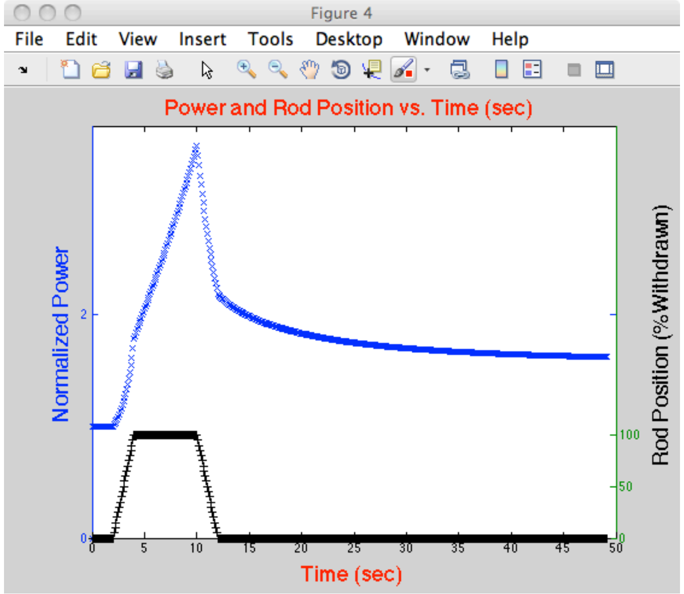


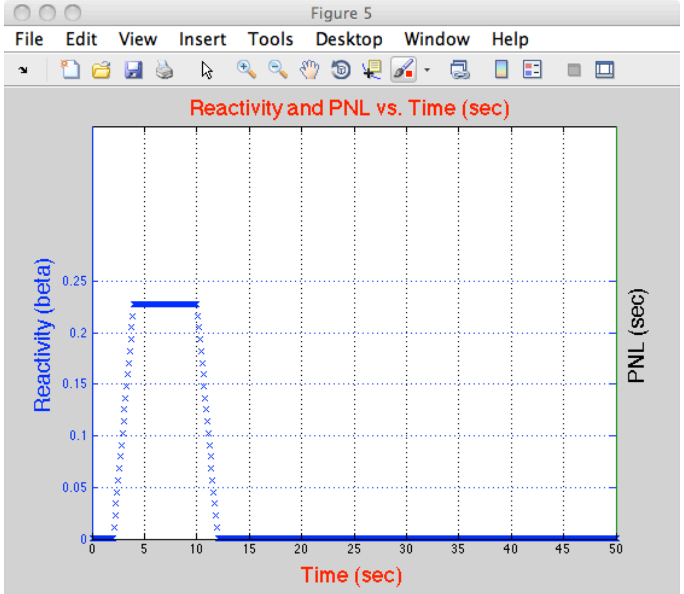


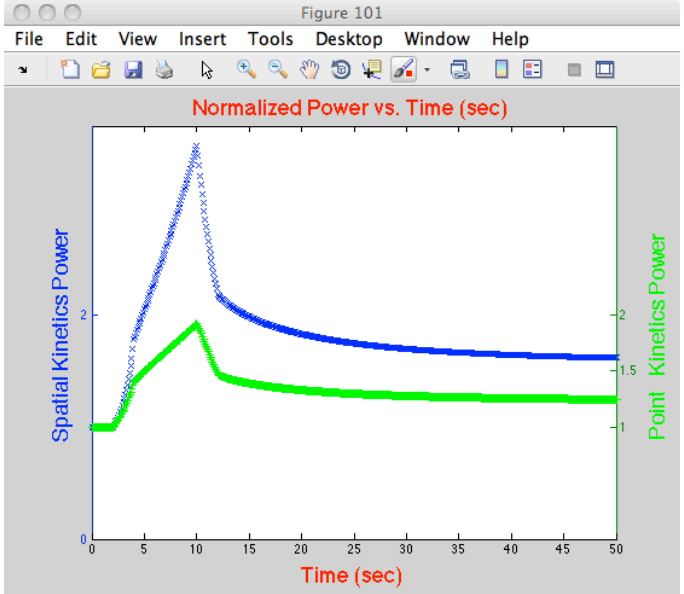


PKE solutions: Rod-In Shape Function

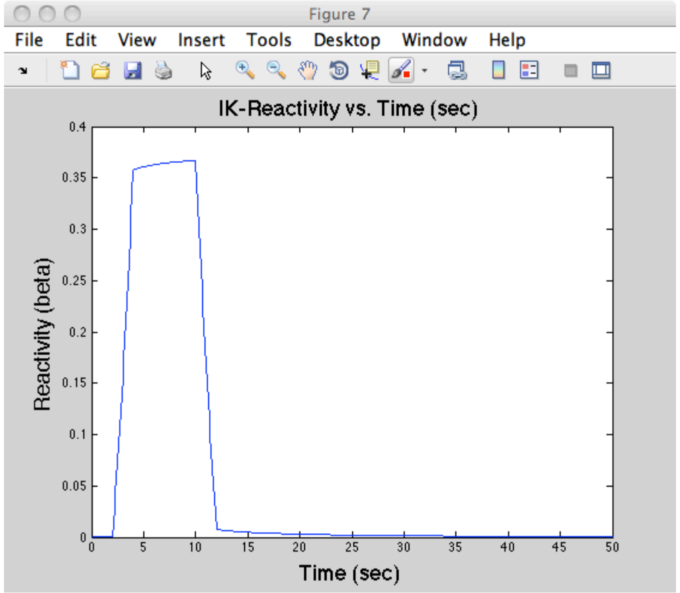


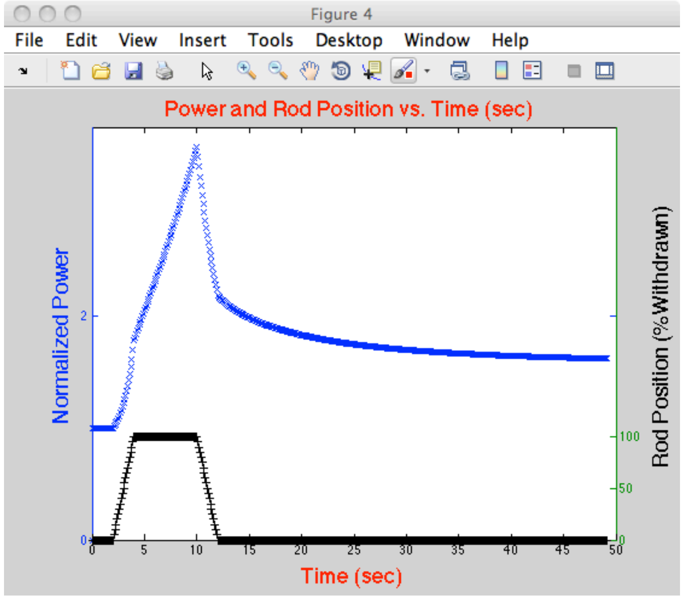


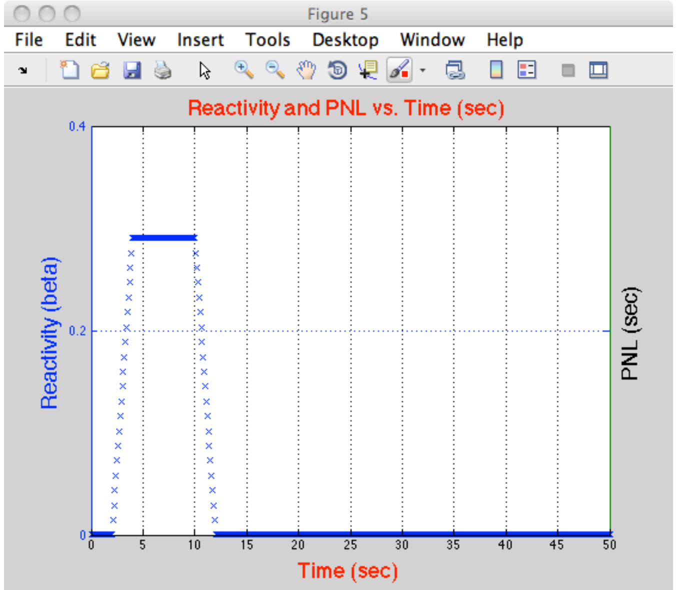


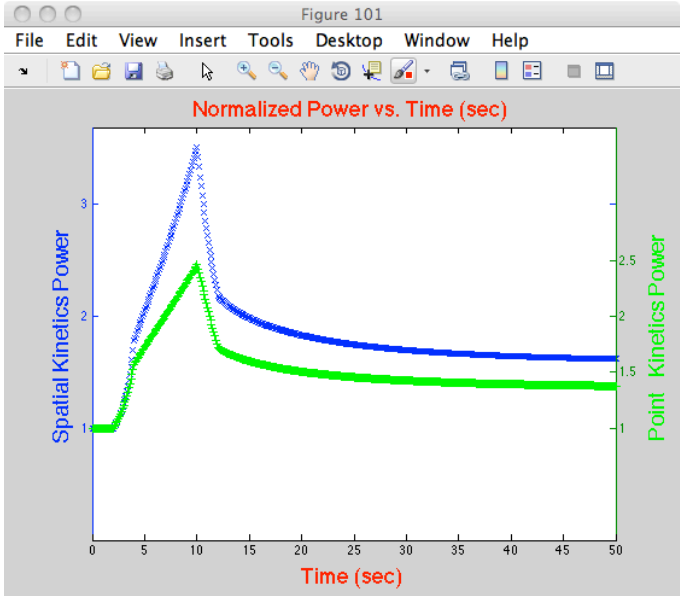


PKE solutions: Rod-In Shape Function & Adjoint

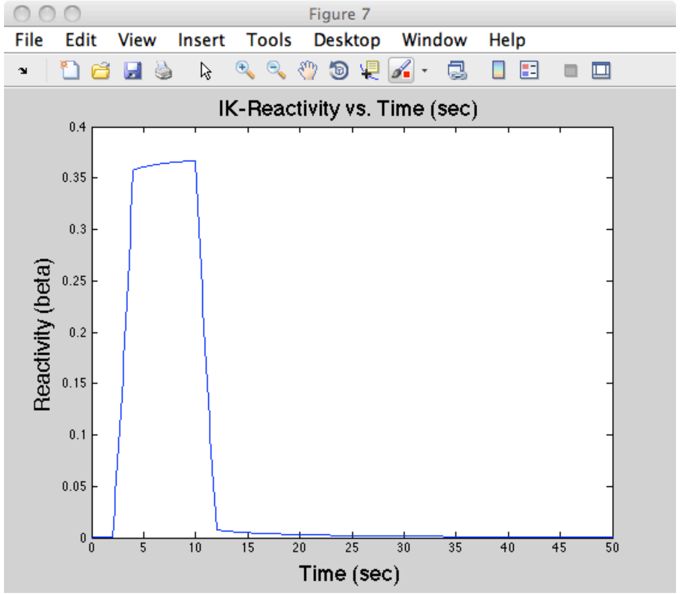


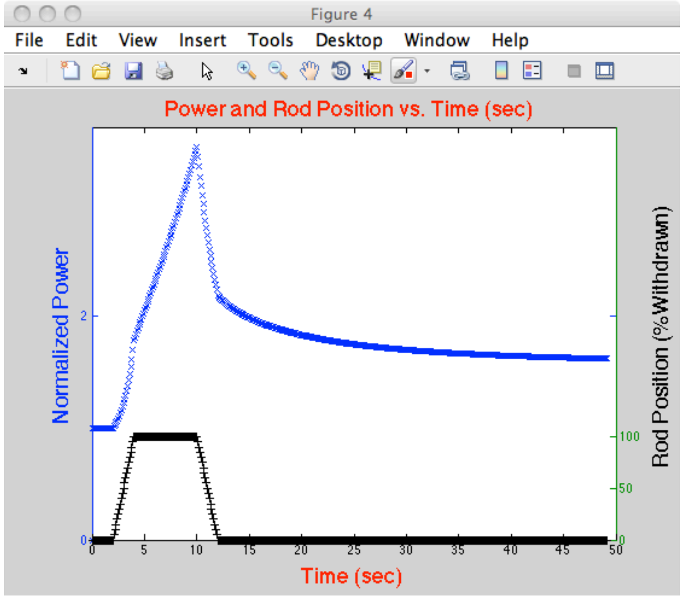


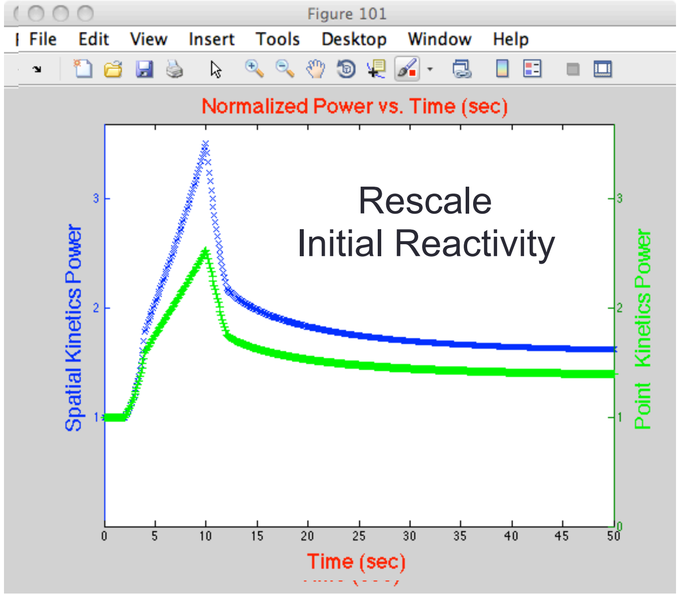




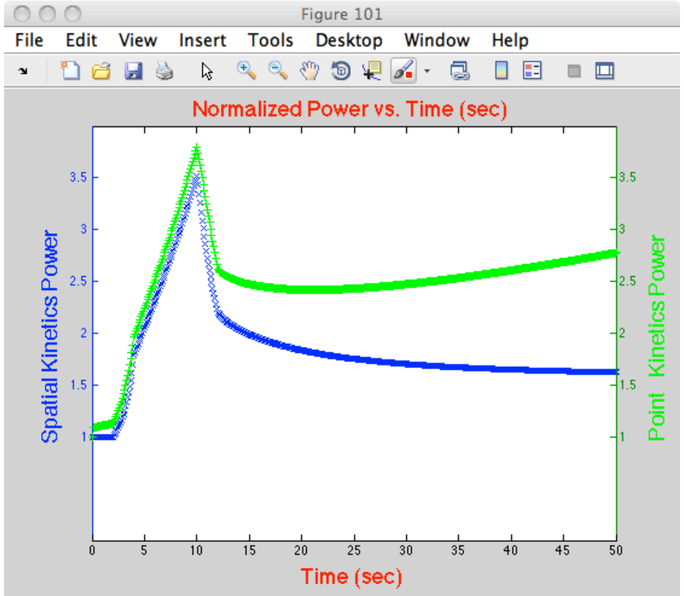
PKE solutions: Rod-Out Shape Function



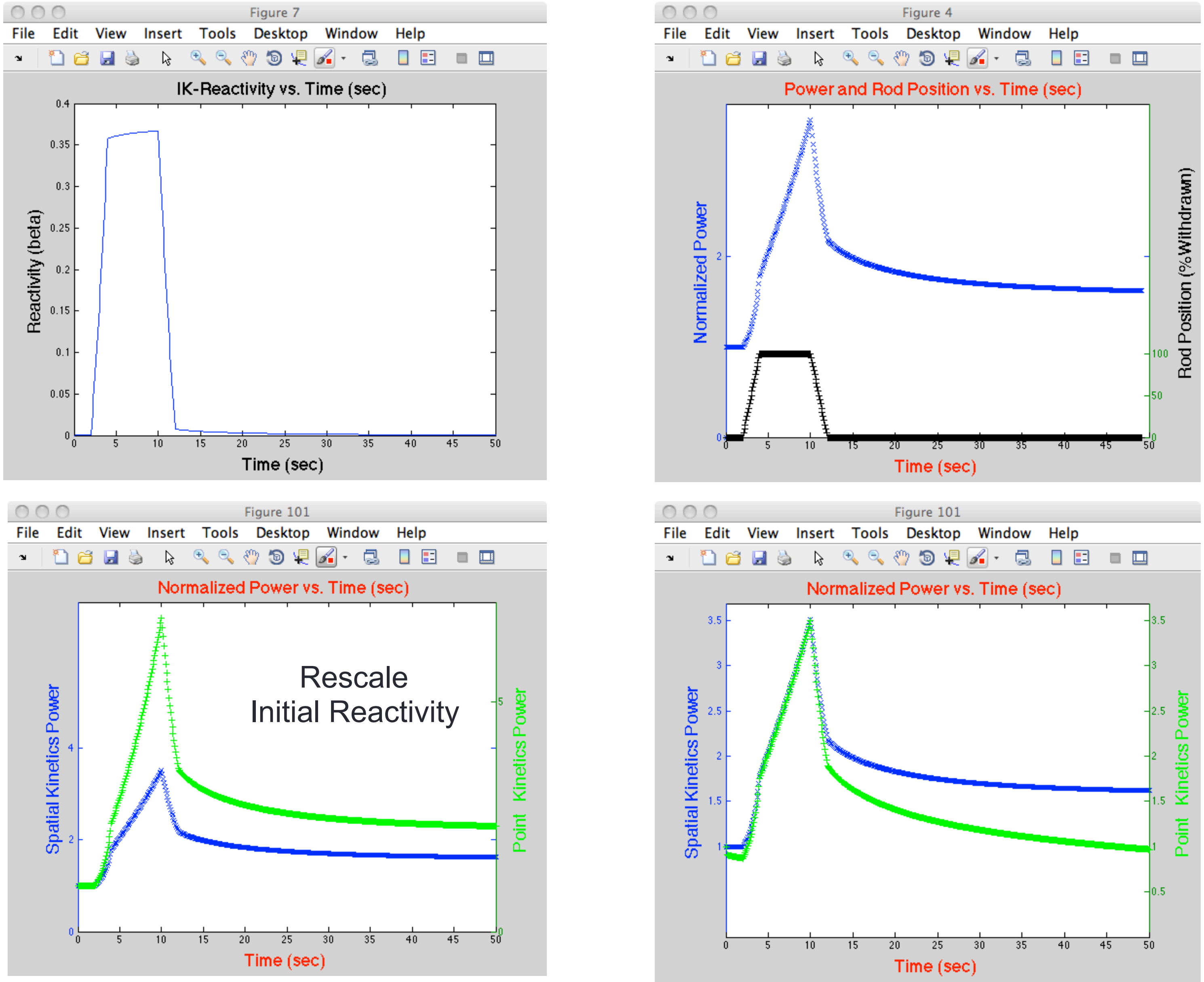




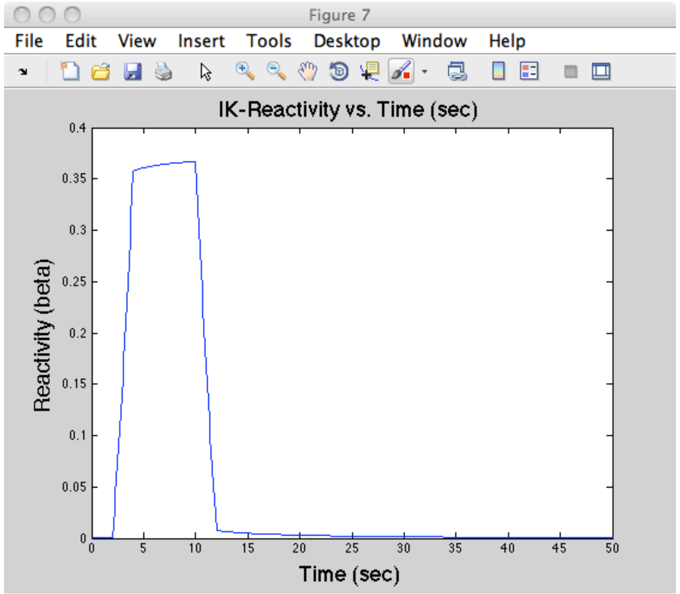
Rescale
Initial Reactivity

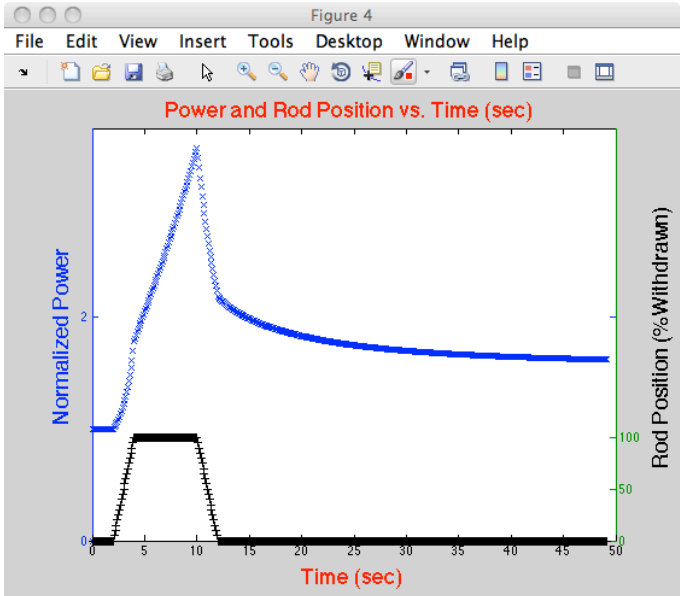


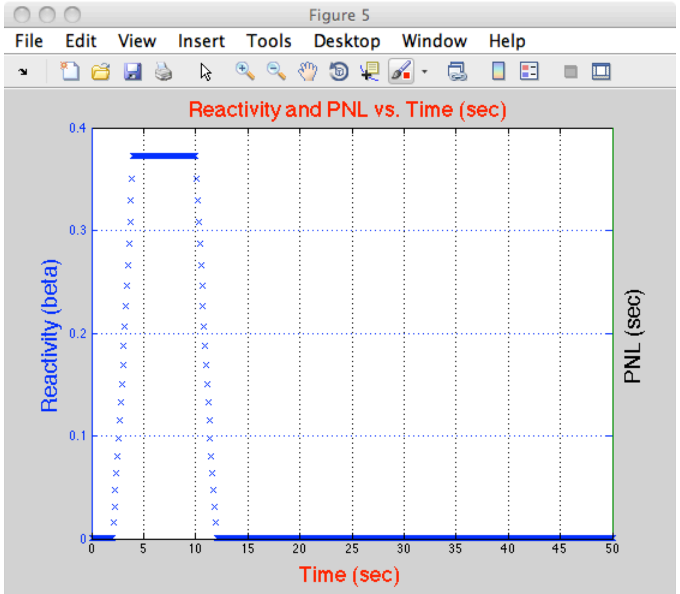
PKE solutions: Rod-Out Shape Function & Adjoint

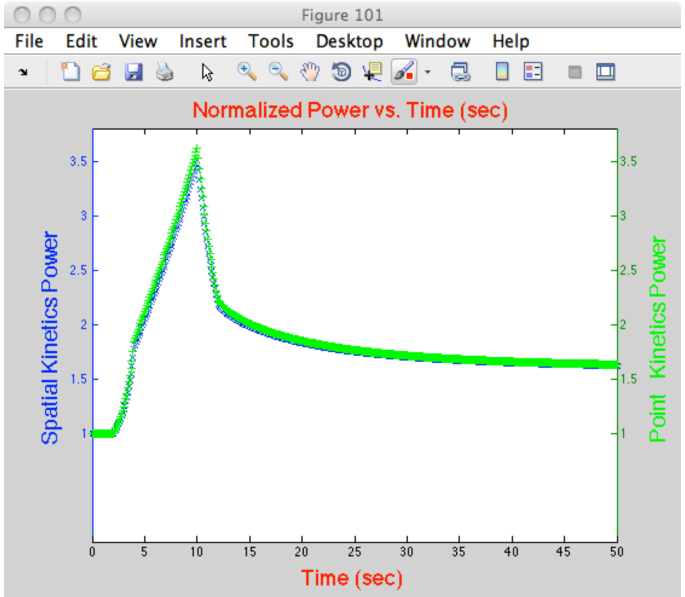


PKE solutions: Interpolation of Shape Functions With Rod Insertion

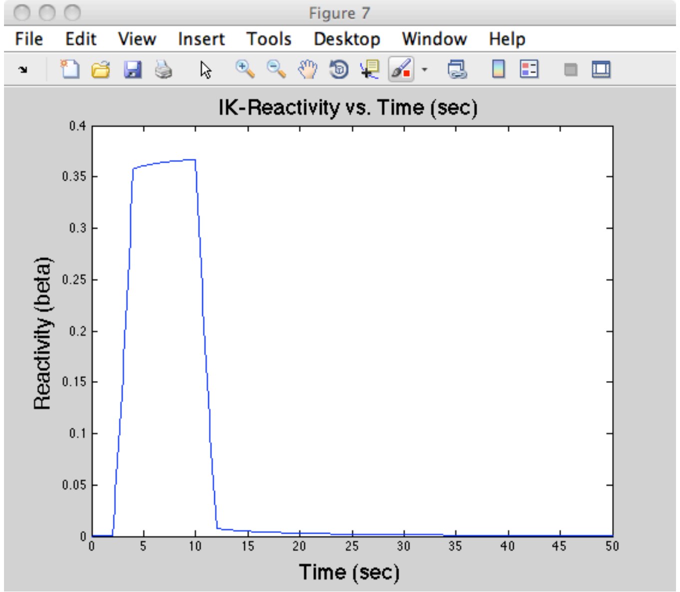


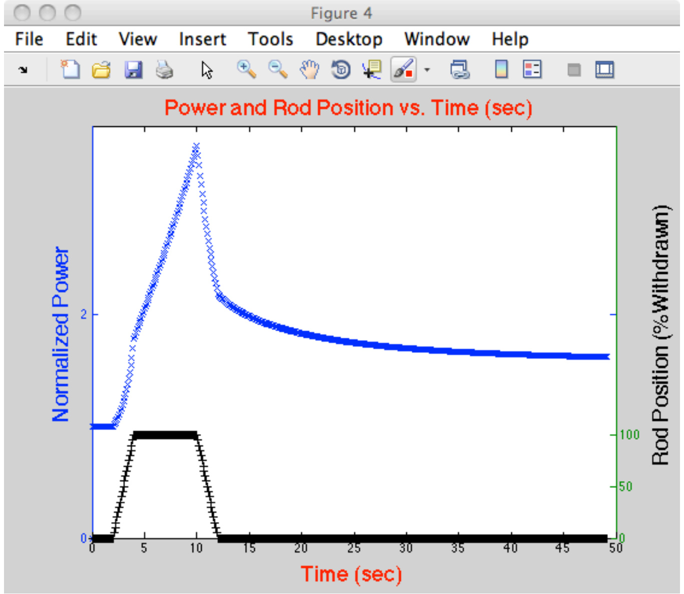


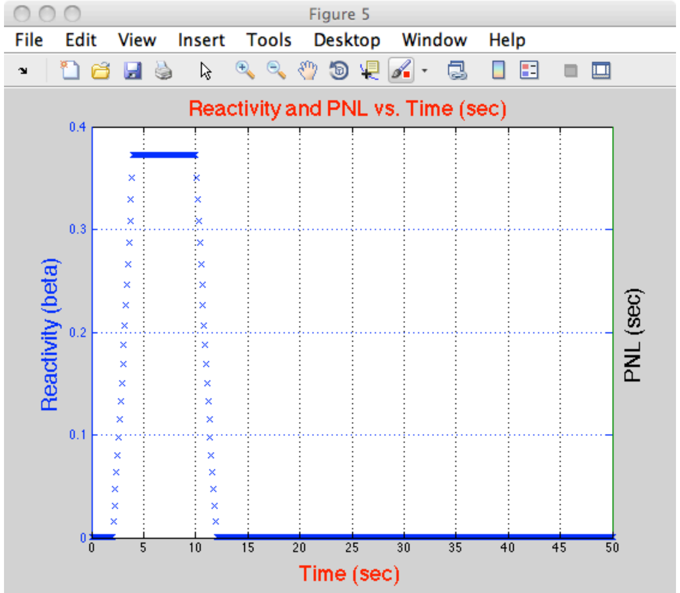


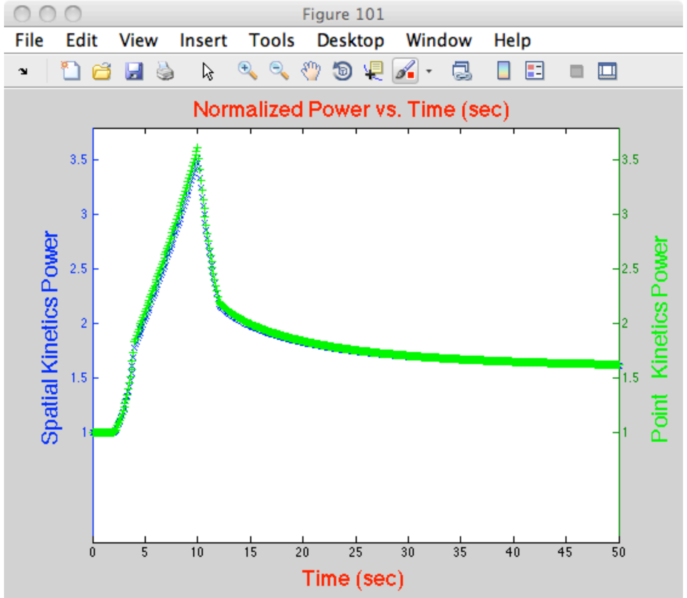


PKE solutions: Interpolation of Shape Function & Adjoint

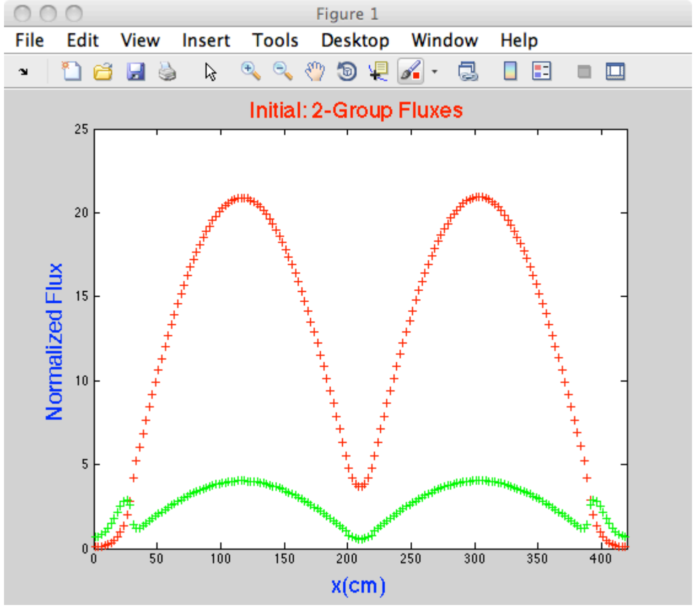


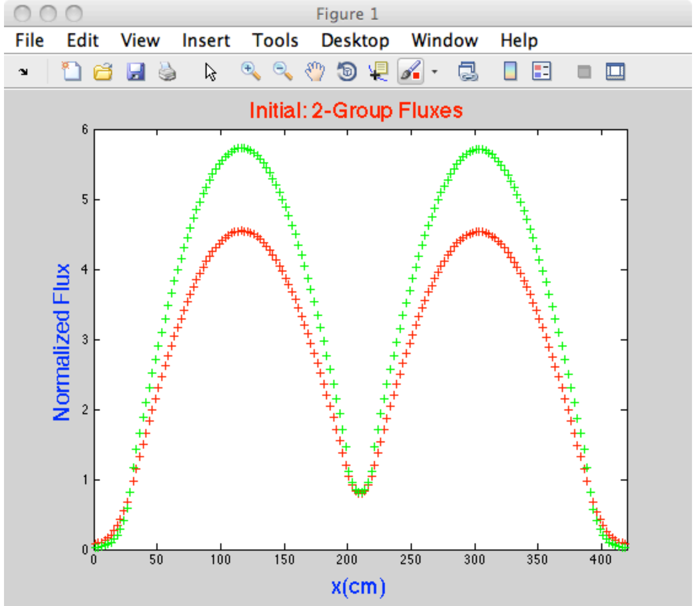


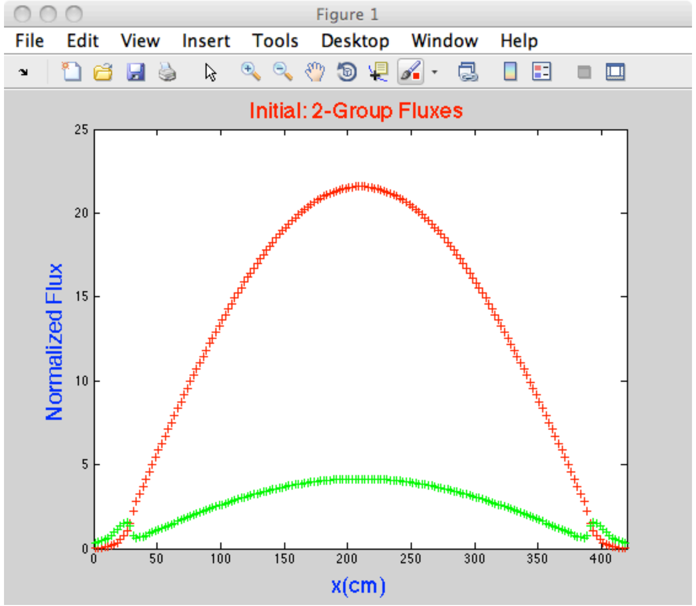


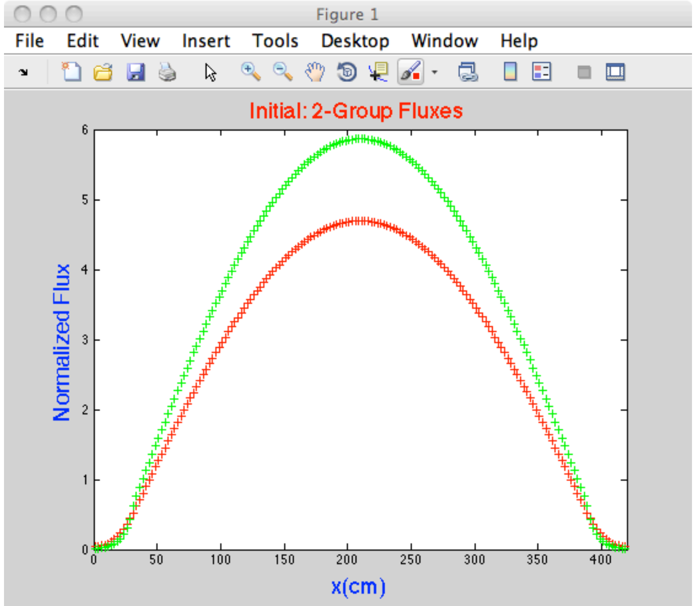


PKE solutions: Rod Ejection/Scram Transient

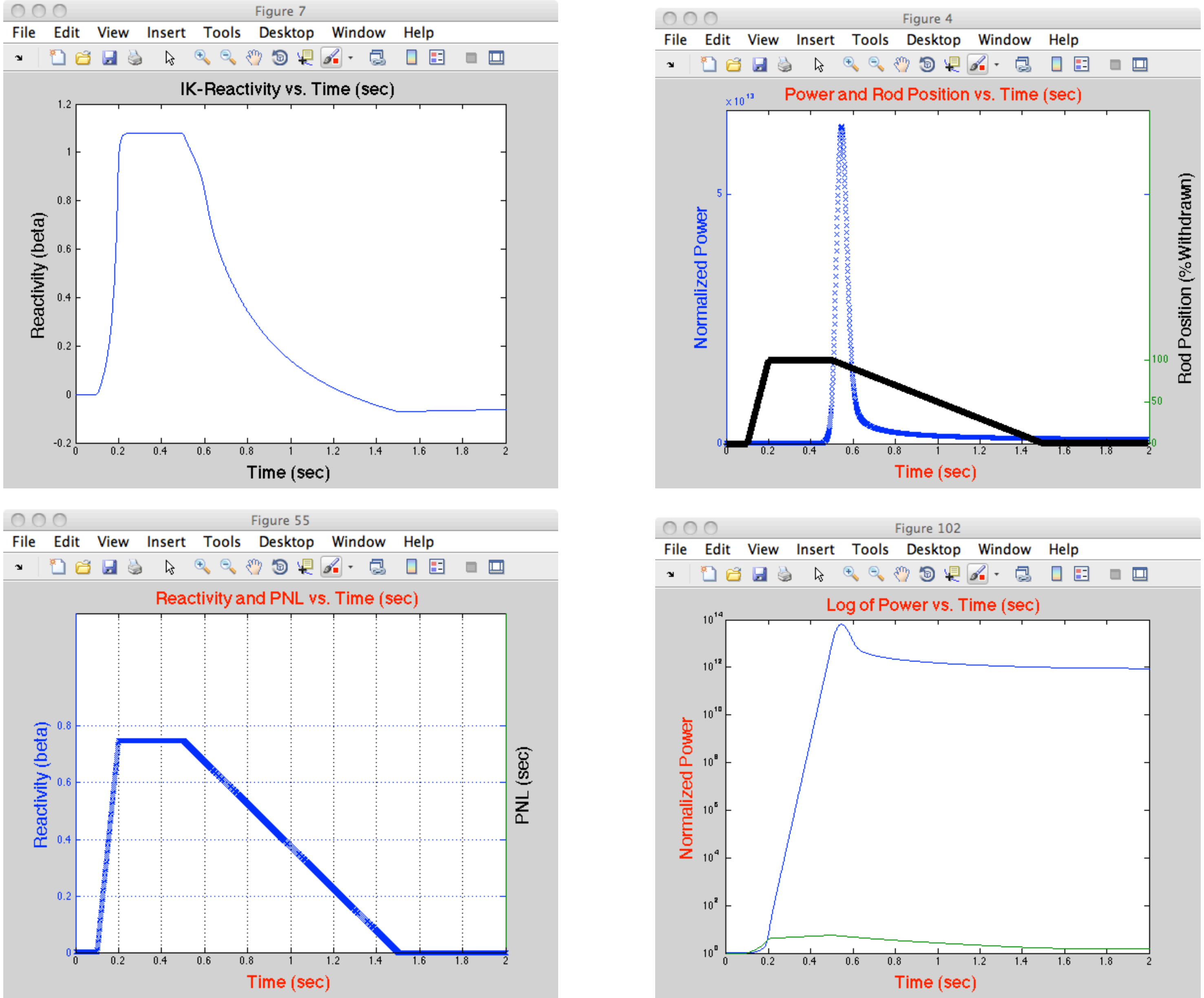




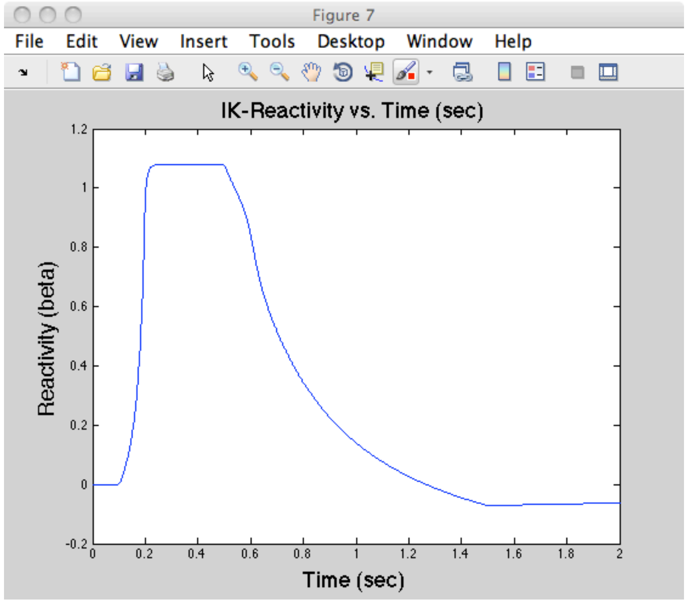


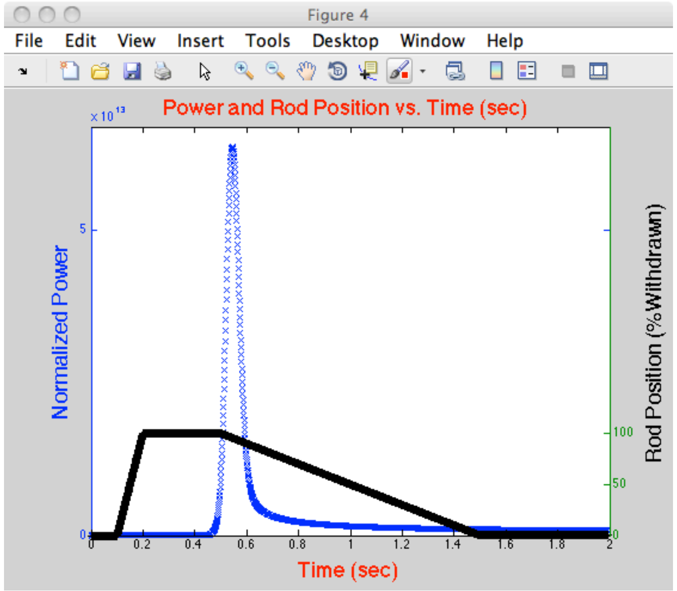


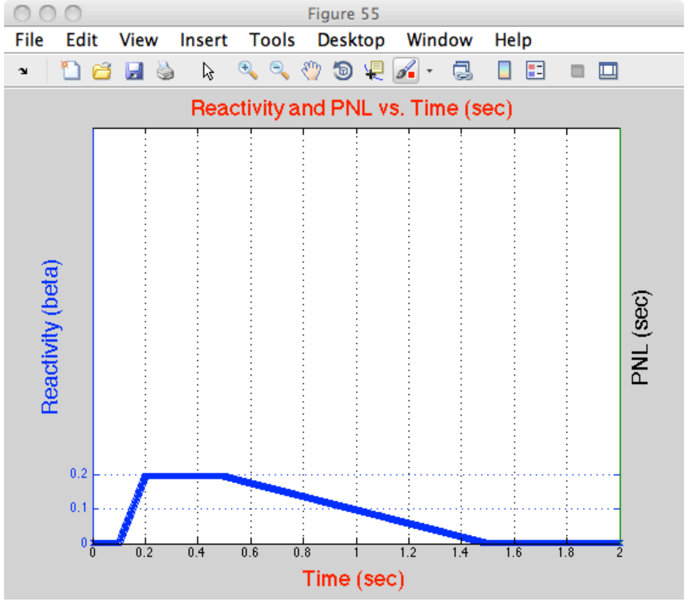
PKE solutions: Rod-in Shape Function

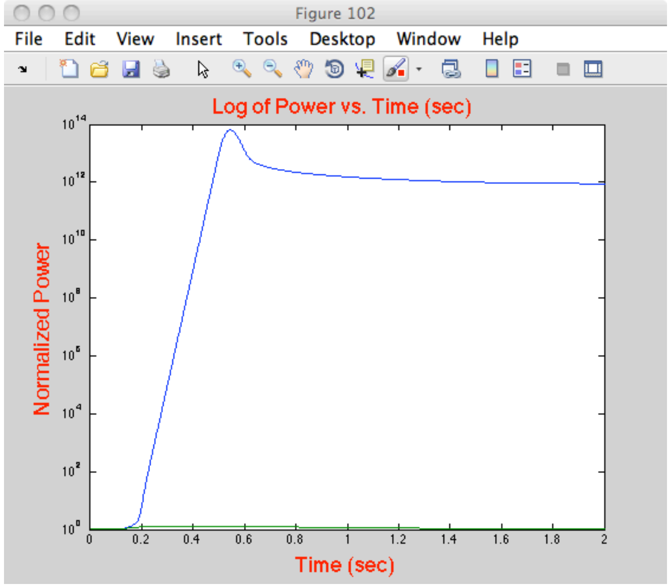


PKE solutions: Rod-in Shape Function & Adjoint

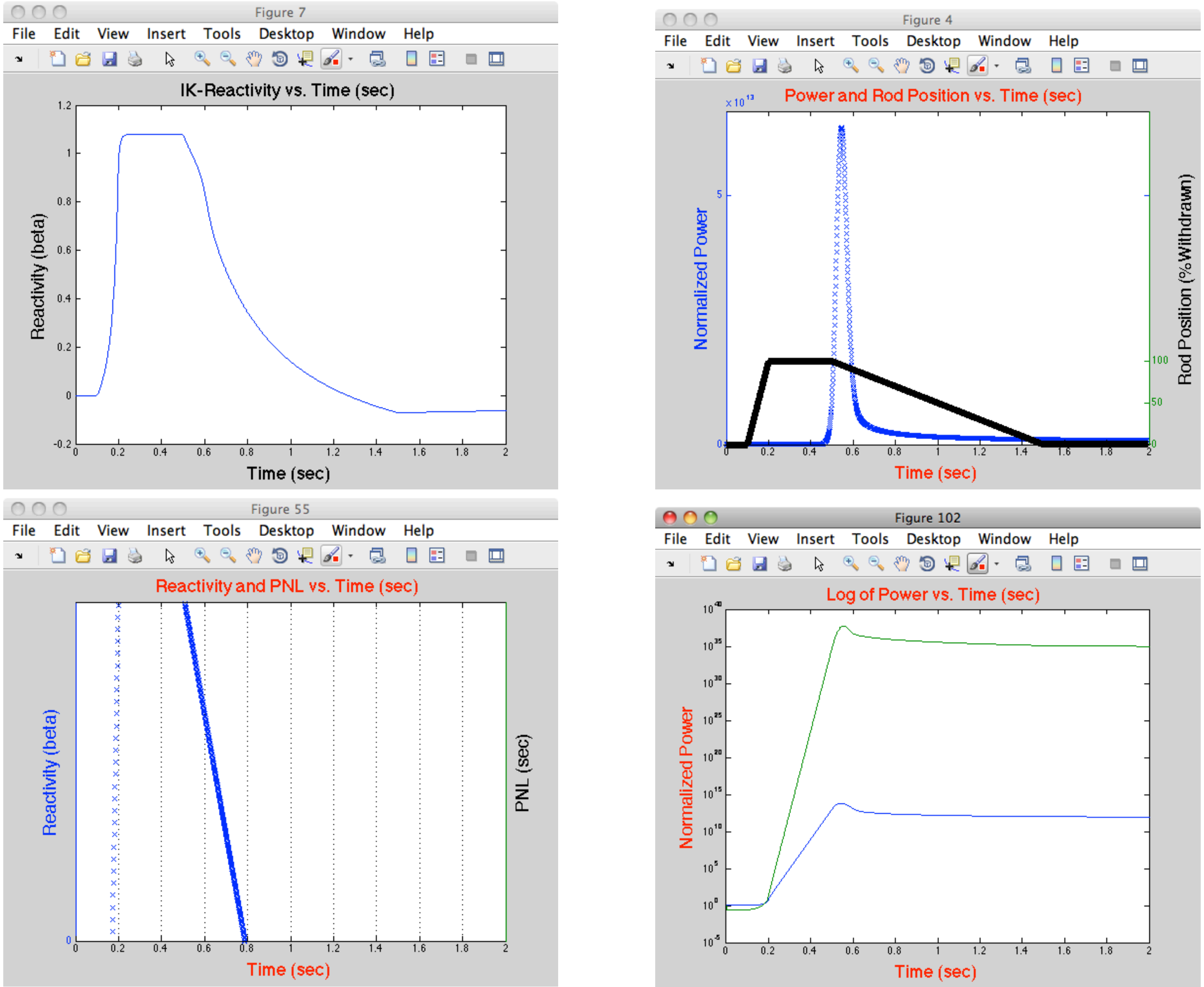




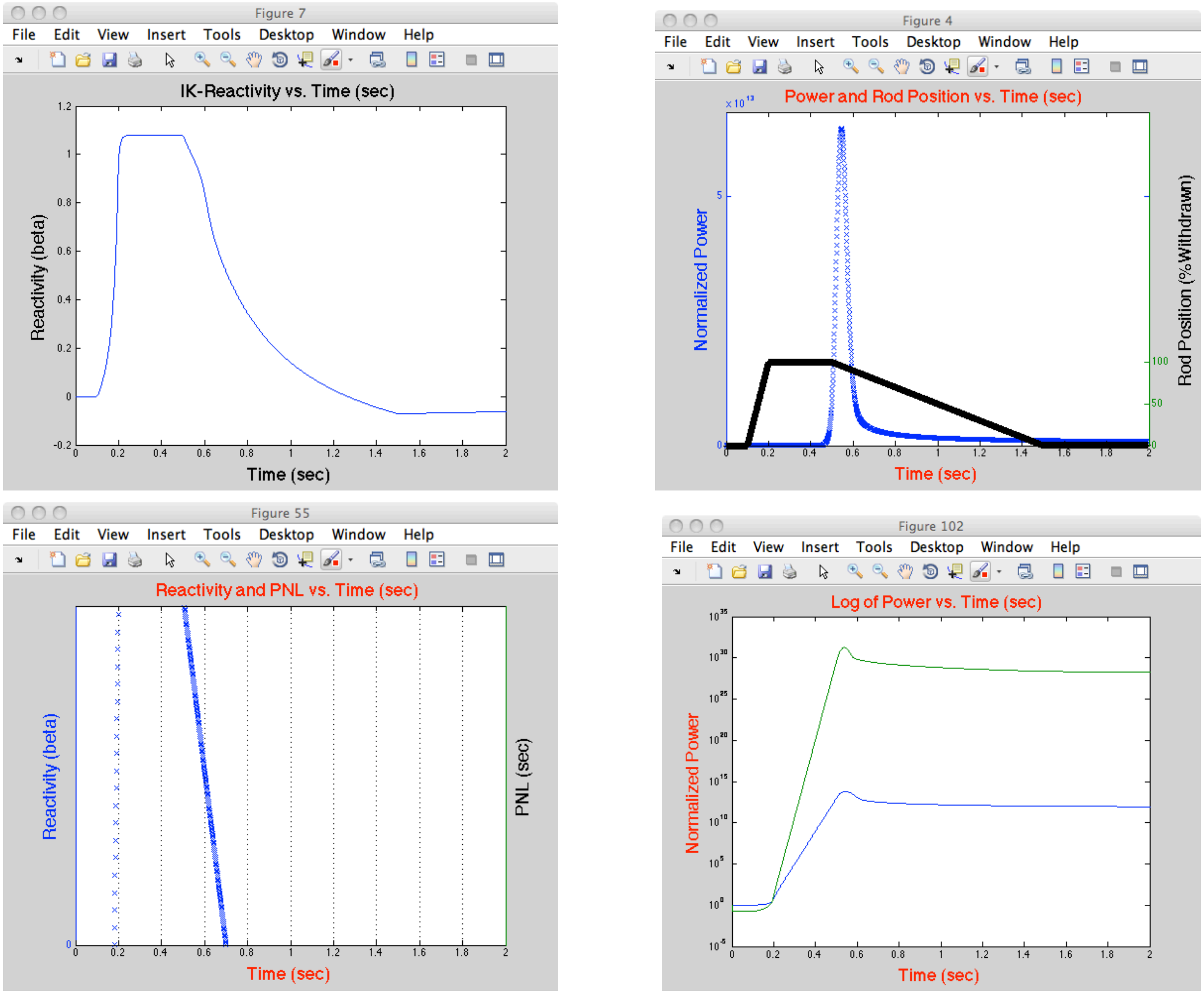




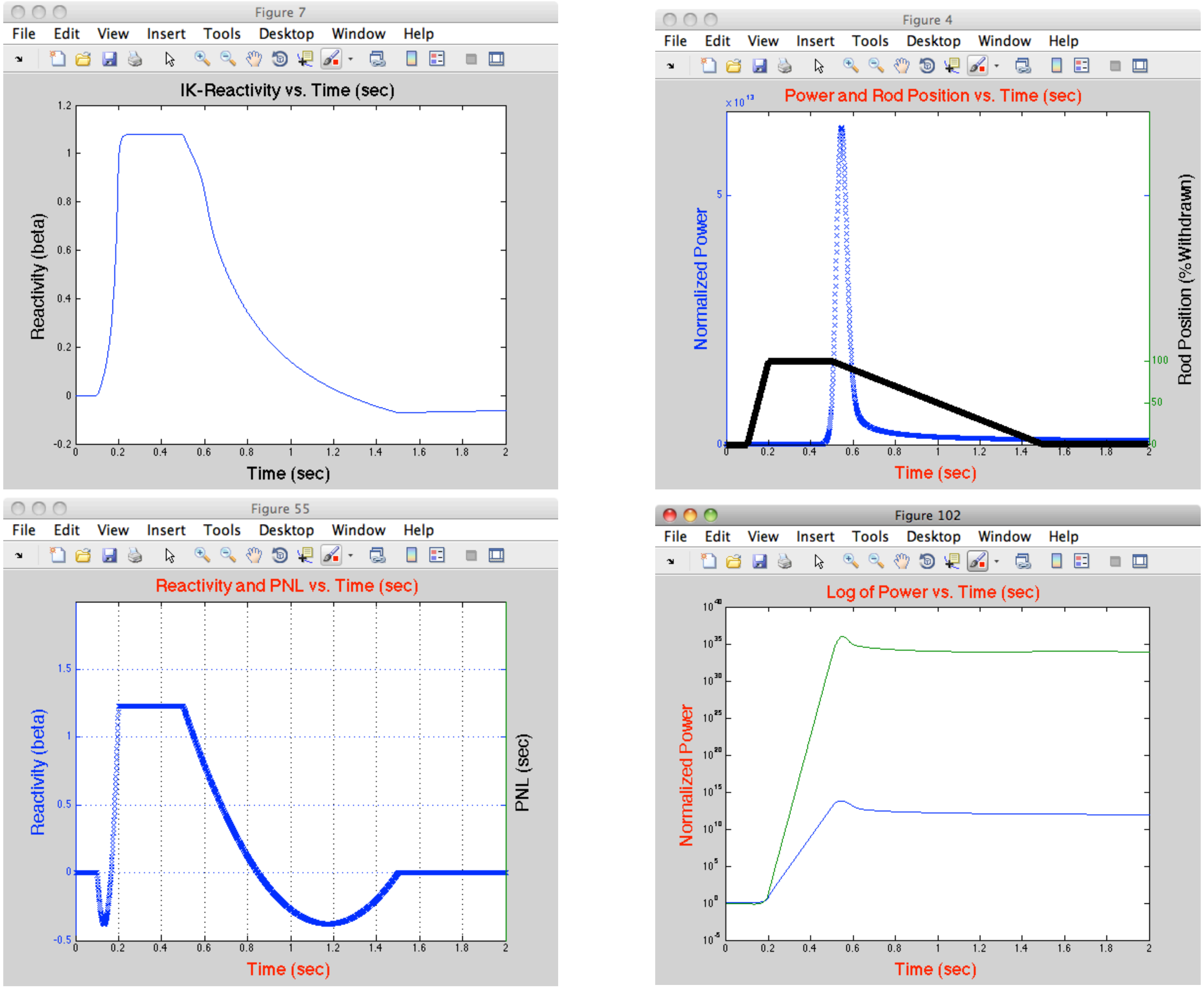
PKE solutions: Rod-out Shape Function



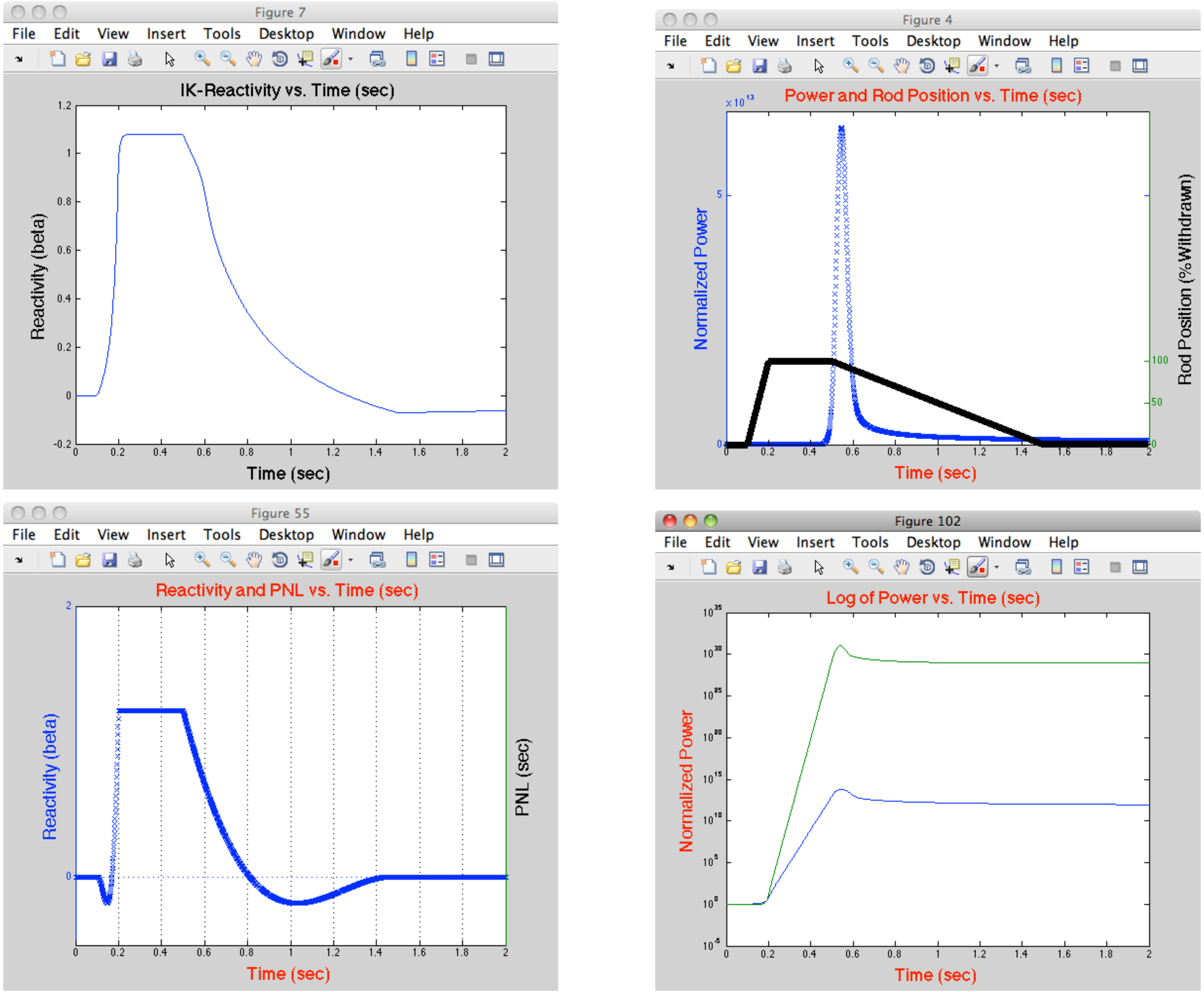
PKE solutions: Rod-out Shape Function& Adjoint



PKE solutions: Interpolation of Shape Functions With Rod Insertion



PKE solutions: Interpolation of Shape Functions & Adjoint



Matrix 2-group PKE Diffusion Equations

$$\begin{bmatrix} \int d\vec{r}\, \frac{1}{v_1} S_1(\vec{r}) & 0 \\ 0 & \int d\vec{r}\, \frac{1}{v_2} S_2(\vec{r}) \end{bmatrix} \frac{d}{dt}\begin{bmatrix} T_1(t) \\ T_2(t) \end{bmatrix} =$$

$$\begin{bmatrix} \int d\vec{r}\, \phi\left[-\hat{\Sigma}_{r,1}(\vec{r},t) + \left[1-\beta(\vec{r})\right]\nu\Sigma_{f,1}(\vec{r},t)\right] S_1(\vec{r}) & \int d\vec{r}\left[\left[1-\beta(\vec{r})\right]\nu\Sigma_{f,2}(\vec{r},t)\right] S_2(\vec{r}) \\ \int d\vec{r}\, \Sigma_{s,1\to 2}(\vec{r},t)\, S_1(\vec{r}) & \int d\vec{r}\left[-\Sigma_{a,2}(\vec{r},t)\right] S_2(\vec{r}) \end{bmatrix} \begin{bmatrix} T_1(t) \\ T_2(t) \end{bmatrix} + \begin{bmatrix} \int d\vec{r}\, \left[\sum_i \lambda_i C_i(\vec{r},t)\right] \\ 0 \end{bmatrix}$$

$$\frac{d}{dt}\left[\int d\vec{r}\;\; C_i(\vec{r},t)\right] = \int d\vec{r}\; \beta_i(\vec{r})\left[\nu\Sigma_{f,1}(\vec{r},t)S_1(\vec{r})T_1(t) + \nu\Sigma_{f,2}(\vec{r},t)S_2(\vec{r})T_2(t)\right] - \lambda_i \int d\vec{r}\;\; C_i(\vec{r},t)$$

Next Time?

Assignment for Next Class

- Solve Pset 3: transient1-D, 2-group finite-difference problems
- Think about how you will compute generalized PKE parameters from your 1-D, 2-group solutions.
- Study the kinetics review paper by Sutton and Aviles (KAPL)
- Start thinking about time-dependent solutions to 2-D, 2-group finite-difference problems.