**NPRE 247, Modeling Nuclear Energy System**

Computer Project 2: Neutron balance eigenvalue/eigenvector problem

Due: see Compass2G

Write a computer program to calculate eigenvalue/eigenvector of a matrix using power iteration method. The main program should do the following:

1. Read cross-section data.
2. Calculate Migration and Fission matrix.
3. Formulate neutron balance as eigenvalue/eigenvector problem: *A = k*
4. Calculate all eigenvalues/eigenvectors using computer software (e.g. MATLAB eig function).
5. Calculate the largest eigenvalue and the corresponding eigenvector using power iteration.
6. Write results to an output file that can later be used to plot the results. The output file should also contain the parameters read from the input file.

Submit a brief report with your results. See separate instructions on the report format.

Problem 1 – Theory

1. Show full derivation of the multi-group neutron balance in infinite space (-10%).
2. Show multi-group Migration and Fission matrix (-10%).

Problem 2 – 2G eigenvalue/eigenvector problem

1. Show Migration, Fission matrix with numerical values (-10%).
2. Calculate all eigenvalues/eigenvectors of the neutron balance analytically by “hand calculation” – show complete derivation and the results (-10%).
3. Calculate all eigenvalues/eigenvectors of the neutron balance using computer software (e.g. MATLAB eig function) – show results (-10%).
4. Calculate the largest eigenvalue and the corresponding eigenvector using power iteration – show results after 2 iterations (-10%).

Problem 3 – 8G eigenvalue/eigenvector problem

1. Show Migration, Fission matrix with numerical values (-10%).
2. Calculate all eigenvalues/eigenvectors of the neutron balance using computer software (e.g. MATLAB eig function) – show results (-10%).
3. Calculate the largest eigenvalue and the corresponding eigenvector using power iteration – show results after 2 iterations (-10%).

2G data (NEACRP L336, homogeneous UO2 composition)

|  |  |  |  |
| --- | --- | --- | --- |
| Group | a | f | chi |
| 1  2 | 0.0092  0.0927 | 0.0046  0.1139 | 1.0000  0.0000 |

Scattering (column 🡪 row)

|  |  |
| --- | --- |
| To\From | 1 2 |
| 1  2 | 1.0000 0.0000  0.0204 2.0000 |

8G data (VENUS-2, UO2 3.3% composition, xs\_pin2-8g-LF)

|  |  |  |  |
| --- | --- | --- | --- |
| Group | a | f | chi |
| 1  2  3  4  5  6  7  8 | 0.0056  0.0029  0.0025  0.0133  0.0473  0.0170  0.0538  0.1788 | 0.0134  0.0056  0.0011  0.0067  0.0220  0.0222  0.0897  0.2141 | 0.3507  0.4105  0.2388  0.0000  0.0000  0.0000  0.0000  0.0000 |

Scattering (column 🡪 row)

|  |  |
| --- | --- |
| To\From | 1 2 3 4 5 6 7 8 |
| 1  2  3  4  5  6  7  8 | 0.1179 0 0 0 0 0 0 0  0.0530 0.1949 0 0 0 0 0 0  0.0301 0.1159 0.5868 0 0 0 0 0  0.0001 0.0005 0.0769 0.8234 0 0 0 0  0 0 0.0019 0.1961 0.8180 0 0 0  0 0 0.0000 0.0050 0.1737 0.6902 0.0023 0  0 0 0 0.0007 0.0246 0.2707 0.8626 0.0275  0 0 0 0.0001 0.0073 0.0550 0.3589 1.9761 |

**Power iteration**

k1 = 1

1 = 1

do i = 1 to MAXIT

i+1 = B i / || B i ||2

ki+1 = (B i+1)T i+1 / (i+1T i+1)

end

Khan Academy has entire course on linear algebra, with sub-section on eigenvalue/eigenvector. For those who need a review of eigenvalue/eigenvector problem, I recommend these two:

1. Conceptual description (refers to previous lectures, some linear algebra will be useful)

<https://www.khanacademy.org/math/linear-algebra/alternate_bases/eigen_everything/v/linear-algebra-introduction-to-eigenvalues-and-eigenvectors>

2. Example for 2x2 matrix

<https://www.khanacademy.org/math/linear-algebra/alternate_bases/eigen_everything/v/linear-algebra-example-solving-for-the-eigenvalues-of-a-2x2-matrix>