MATLAB/SIMULINK PLATFORM FOR SIMULATION OF CANDU REACTOR CONTROL SYSTEM

Hooman Javidnia and Jin Jiang

Department of Electrical and Computer Engineering
The University of Western Ontario, London, Ontario, Canada, N6A 5B9
{hjavidn2,jjiang}@uwo.ca

Abstract

In this paper a simulation platform for CANDU reactors' control system is presented. The platform is built on MATLAB/SIMULINK interactive graphical interface. Since MATLAB/SIMULINK are powerful tools to describe systems mathematically, all the subsystems in a CANDU reactor are represented in MATLAB's language and are implemented in SIMULINK graphical representation. The focus of the paper is on the flux control loop of CANDU reactors. However, the ideas can be extended to include other parts in CANDU power plants and the same technique can be applied to other types of nuclear reactors and their control systems. The CANDU reactor model and xenon feedback model are also discussed in this paper.

1. Introduction

Most of the control systems in use for nuclear reactors were designed during 60s based on classical control theories. These designs are based on frequency response methods and time domain methods for single input single output (SISO) systems. Designers of these systems tried to break down the complicated loops into simpler ones and make the design process easier to handle. However, loop interactions are difficult to deal with.

Simulation codes play a major role in design and analysis of nuclear power plants. Atomic Energy of Canada, Ltd. (AECL), Ontario Power Generation (OPG), and other utility companies have developed simulators and codes to meet their needs. Some of these codes are designed to perform specific tasks, others are simulators of larger scope for training purposes, which are intended to simulate the whole power plant or a subset of its processes.

Nuclear power plant simulators are deployed either as full-scope training simulators or in a smaller version as desktop simulators. Owing to the complexity of nuclear power plants, and limitations on computational power and storage of computers, simulators used to require dedicated hardware to run in real-time. The recent developments of computer technology has made it possible to develop computer codes and simulators that can run on conventional desktop computers or on a cluster of computers.

Power plant simulators for existing CANDU designs are developed at OPG and are available for training and analysis purposes. These simulators are behind all the full-scope training simulators and are also available in desktop versions. The desktop version requires an alpha-processor machine to run in real-time. Cassiopeia Technologies Inc (CTI) has developed PC-based desktop simulators for generic type NPPS such as PWR, BWR, CANDU 9, and VVER. These simulators are available through International Atomic Energy Agency (IAEA). The simulators are based on CASSIM, which is the simulation development environment developed at CTI [1]. AECL and other utility companies have done extensive work in this field, but most of their work is not reported in open literature and are considered proprietary.

One of the major difficulties that the early designers faced is still troublesome today. This is the lack of a unified simulation environment. In a unified simulation environment, the control system engineers could test their ideas and designs before they implement into a real reactor control system. Historically, many computer codes have been developed to solve different kinds of problems that arise in nuclear reactor engineering. The issue is that the communication between these codes has never been trivial. The computer codes that are sophisticated in solving transient characteristics of a reactor, are not always equipped with control system design tools and vice versa. Simulators such as CTI simulators provide a unified environment for simulation, but the core of the simulation is based on a new language which is unfamiliar to most engineers and difficult to modify.

In this paper a unified approach towards simulation of nuclear reactors and their control systems has been developed. To achieve this goal MATLAB and its interactive graphical modeling environment SIMULINK are used. The effort is paid to CANDU reactors, although the general idea can be applied to other types of power reactors with appropriate modifications. This is an extension of the work reported in [2], with the objective of improving the models and making them more concise. A main advantage of this approach is that it is based on MATLAB language and the blocks are built in SIMULINK which are familiar to most of control engineers and simulation scientists.

2. Reactor Regulating System

The focus of this paper is on the flux control loop of a CANDU reactor which controls the reactor power throughout its allowed range of operation. Reactor power is controlled by various reactivity devices. Reactor control in CANDU is computerized and is implemented in the Reactor Regulating System (RRS) program in the plant control computers: Direct Digital Control Computers (DCC) system consisting of two computers (DCCX and DCCY). The two computers are identical and each one can control the reactor independently. The advantage of redundancy is the increased reliability and availability of the control system computers.

The reactivity control devices in CANDU reactors are:

- Light water Liquid Zone Controllers (LZC)
- Adjuster Rods (AR)
- Mechanical Control Absorbers (MCA)

On-power re-fuelling for compensating long term reactivity effects, and liquid moderator poison for providing negative reactivity can also be used as reactivity control mechanisms.

Figure 1 shows a simplified version of the reactor control system. For the purpose of this study, the block diagram is a good representation of the flux control loop in a CANDU reactor. Among several reactivity control devices which are present, only LZC is modeled explicitly. The reactivity worth of all other devices are lumped into one single reactivity source. This source is shown as “Other Reactivity Devices” in Figure 1.

The control algorithms hosted on DCCX and DCCY lie in the heart of the flux control loop. There are many routines on the control computers, but in this paper only a few crucial ones in the flux control loop are cosidered.

3. Vectorization of Models

An important feature of all the models is *vectorization*. Since the reactor is divided into 14 zones, other elements of the control system also work in a dimension of 14 or higher. MATLAB/SIMULINK is very

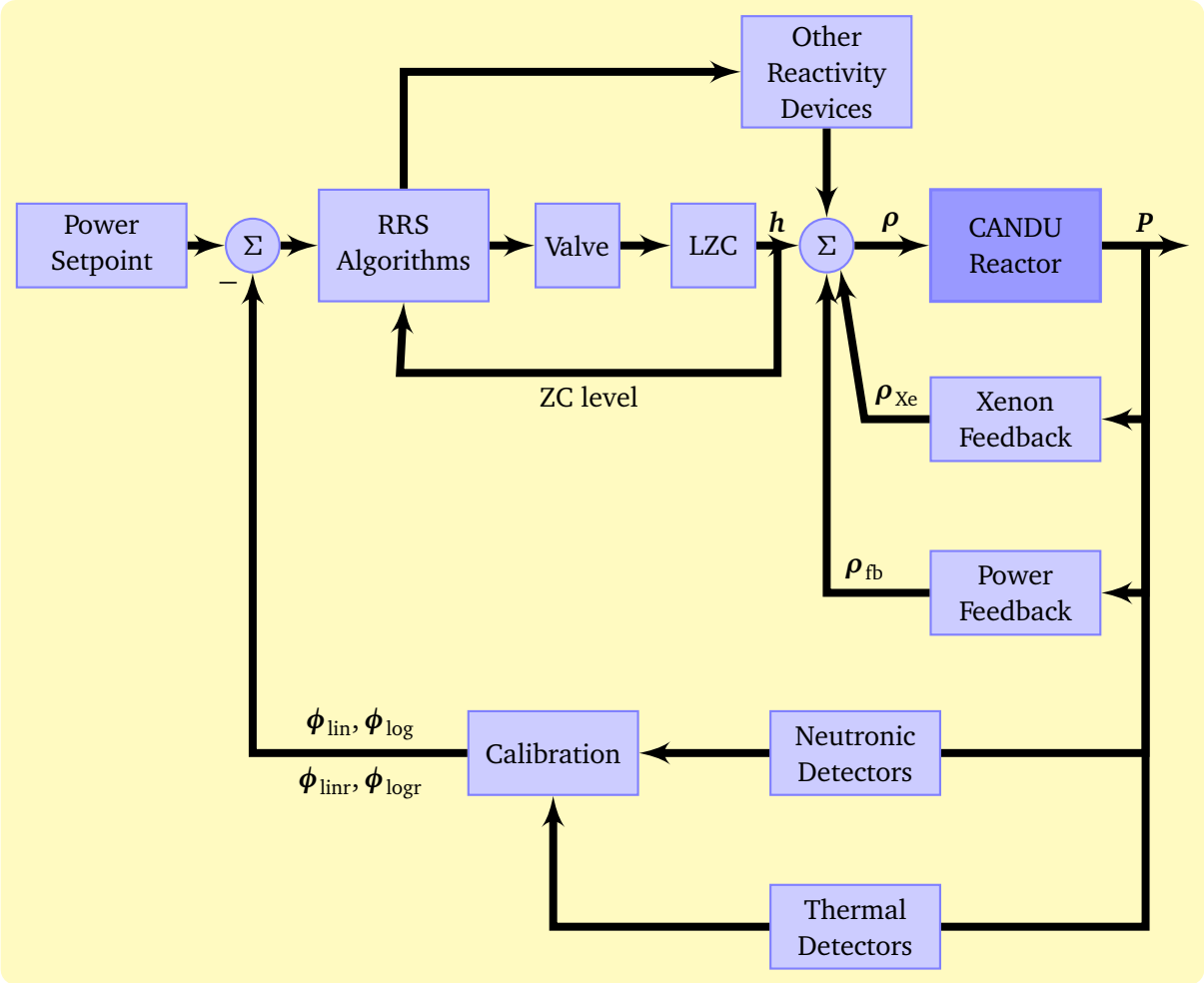

Figure 1: Simplified block diagram of CANDU reactor control system

proficient at handling vectors and matrices. This is one of the major strongholds of MATLAB. Using this feature, it is possible to implement the models in a very compact and concise form. The vectorized implementations are more efficient and are easier to read and understand in the graphical environment of SIMULINK.

4. CANDU Reactor

Nuclear reactors are the heart of every nuclear power plant. The most commonly used technologies in nuclear power plants are Boiling Water Reactors (BWR), Pressurized Water Reactors (PWR), and CANada Deuterium Uranium (CANDU) reactors. All the nuclear power plants in Canada have a CANDU type reactor in their core with minor differences in their design.

The model of a CANDU reactor in this paper is a nodal model adopted from the work done by A. P. Tiwari and colleagues reported in [3] and a series of articles such as [4]. The modeling process starts from a two-group diffusion description of the reactor dynamics. The usual assumptions of multigroup diffusion theory have been applied along with some other assumptions. For detailed discussion of the assumptions on diffusion theory, interested readers are referred to [5] and [6].

The equations for epithermal and thermal neutron group are given as

$$\frac{1}{\nu_1}\frac{\partial \phi_1}{\partial t} = \boldsymbol{\nabla}\cdot D_1\boldsymbol{\nabla}\phi_1 - \Sigma_{a1}\phi_1 + (1-\beta)(\nu\Sigma_{f1}\phi_1 + \nu\Sigma_{f2}\phi_2) - \Sigma_{12}\phi_1 + \sum_{h=1}^{m_d}\lambda_h\tilde{C}_h \tag{1}$$

$$\frac{1}{\nu_2}\frac{\partial \phi_2}{\partial t} = \boldsymbol{\nabla}\cdot D_2\boldsymbol{\nabla}\phi_2 - \Sigma_{a2}\phi_2 + \Sigma_{12}\phi_1\,, \tag{2}$$

where epithermal (fast) and thermal (slowed down) groups are represented by subscripts 1 and 2. $\phi(\boldsymbol{r},t)$ stands for the neutron flux. $\tilde{C}_h(\boldsymbol{r},t)$ is the number density of delayed neutron precursors[1]. Σ_a and Σ_f stand for absorption and fission cross sections, respectively. Σ_{12} is the scattering cross section from fast group to thermal group. All the cross sections are functions of position $\boldsymbol{r}$. λs are the decay constants of the delayed neutron precursor groups and β is the total fractional yield of delayed neutrons. The speed of the neutron in each group is denoted by ν_1 and ν_2. This form of the diffusion equations is very common when studying a reactor dynamics with two energy groups. The assumptions made here are similar to the ones made for AECL's RFSP code [7], [8], and many other reactor physics codes. The delayed neutron precursor concentrations evolve according to

$$\frac{\partial \tilde{C}_h}{\partial t} = \beta_h(\nu\Sigma_{f1}\phi_1 + \nu\Sigma_{f2}\phi_2) - \lambda_h\tilde{C}_h \qquad h = 1,2,\ldots,m_d\,. \tag{3}$$

In nodal approximation, the 3D space is divided into small boxes or cells. Equations (1–3) are averaged over the volume of cell i. Averaging gives rise to new fluxes Φ_1 and Φ_2, and all the neutronic properties are averaged over the volume of the cell accordingly.

$$\Phi_{ki}(t) \triangleq \frac{1}{V_i}\int_{V_i}\phi_k(\boldsymbol{r},t)\,\mathrm{d}V \qquad k = 1,2\,. \tag{4}$$

Note that the fluxes defined in Equation (4) are only function of time. Subscript k stands for the fast and thermal group and i is an index to the cell number. It is assumed that Φ_{ki} is the value of the flux at the center of the cell i. The cross sections are treated in a similar fashion; for example, the absorption cross section of fast and thermal groups are flux-weighted according to

$$\Sigma_{aki} \approx \frac{\int_{V_i}\Sigma_{ak}(\boldsymbol{r})\phi_k(\boldsymbol{r},t)\,\mathrm{d}V}{\int_{V_i}\phi_k(\boldsymbol{r},t)\,\mathrm{d}V} = \frac{\int_{V_i}\Sigma_{ak}(\boldsymbol{r})\phi_k(\boldsymbol{r},t)\,\mathrm{d}V}{\Phi_{ki}(t)}\,. \tag{5}$$

The main source of the problem is the leakage term (the first term on RHS of Equations (1) and (2)). In general, the leakage term, if integrated over the volume of the cell can be written as [6]

$$\frac{1}{V_i}\int_{V_i}\boldsymbol{\nabla}\cdot D_k(\boldsymbol{r})\boldsymbol{\nabla}\phi_{ki}(\boldsymbol{r},t)\,\mathrm{d}V \triangleq -\alpha_{ki}\Phi_{ki} + \sum_{j=1}^{J}\alpha_{kij}\Phi_{kj}\,, \tag{6}$$

where, i and j are the indexes of the cells, J is the number of neighbouring cells to cell i, and k is the index for fast and thermal group ($k = 1,2$). α_{1ij} and α_{2ij} are the coupling coefficients between the cells for fast and thermal neutron energies. The diagonal elements of the coupling coefficient matrix can be calculated from the relation

$$\alpha_{kii} = \sum_{j=1}^{J}\alpha_{kij} \tag{7}$$

[1]Photo-neutrons get the same treatment as delayed neutrons

Using Equation (6), it is possible to simplify Equations (1) and (2), because it will remove all the second order partial derivatives. To apply this technique, one has to address another question. How to calculate the coupling coefficients? The coupling coefficients between mesh cells are determined by the nodalization scheme which is chosen for the geometry of the core, and the finite-difference methods used. Coupling coefficients are functions of the nodalization geometry and finite-difference scheme that one chooses to discretize the derivatives [6].

Different expressions for coupling coefficients have been proposed in the literature. For a large PHWR which is very similar to CANDU reactors in its design, the coupling coefficients between nodes i and j are given as [4]

$$\alpha_{ij} = \frac{D v \ell A_{ij}}{d_{ij} V_i}, \tag{8}$$

where D is the diffusion constant after collapsing fast and thermal neutron fluxes in each node according to $1\frac{1}{2}$ group theory [9], v is the thermal neutron speed (again after collapsing the groups), d_{ij} is the center-to-center distance of nodes i and j, ℓ is the prompt neutron lifetime, A_{ij} is the interface area between nodes i and j and V_i is the volume of the node i. Each node of the nodalization scheme is equivalent to one of the zones of the CANDU reactors and hence there are only 14 nodes in this scheme. Coupling between the zones which are not immediate neighbors is considered to be zero.

The spatial coupling coefficients between mesh cells are not unique and depend on such factors as nodalization scheme and finite difference method. Reference [9] gives a different expression for coupling coefficients. For a slightly different coupled nodal kinetics model of the core, another expression is given to represent the coupling coefficients between the nodes in [10]. In all these studies it has been assumed that the coupling coefficients are constant during the transients and they are treated as parameters once they are determined. If the flux shape changes in the reactor owing to a perturbation in configuration of the core, the coupling coefficients determined from an unperturbed flux shape will no longer be accurate and they introduce error to the flux transient tilting [11].

The flux of the node i after collapsing the fluxes of fast and thermal groups is $\Phi_i(t) = \Phi_{1i}(t) + \Phi_{2i}(t)$, and all the macroscopic cross sections are collapsed using the same principle. The power of node (zone) i can be calculated by

$$P_i(t) = E_f \Sigma_{fi} V_i \Phi_i(t), \tag{9}$$

where, E_f is the energy released per fission. Equation (3) can be simplified further, if a new variable for delayed neutron precursor concentrations is defined similar to the power of each node. This modified precursor concentration is given as

$$C_{hi}(t) = E_f \Sigma_{fi} \nu_i V_i \tilde{C}_{hi}(t), \tag{10}$$

where, ν_i is the neutron speed in node i (after collapsing the groups) and $\tilde{C}_{hi}(t)$ is the precursor concentration after it is being averaged over the volume of the node. h is an index to the number of delayed neutron groups ($h = 1, \ldots, m_d$). It should be noted that $C_{hi}(t)$ is measure in the same units as as $P_i(t)$ (W) after this change of variables.

Substituting the average flux of the zone with Equation (9) and the average delayed neutron precursor concentrations with Equation (10), a set of coupled kinetics equations for zones of the reactor in a form similar to point kinetics equations can be obtained as follows:

$$\frac{dP_i}{dt} = (\rho_i - \beta)\frac{P_i}{\ell} - \sum_{j=1}^{Z} \alpha_{ij} \frac{P_i}{\ell} + \sum_{j=1}^{Z} \alpha_{ji} \frac{P_j}{\ell} + \sum_{h=1}^{m_d} \lambda_h C_{hi} \tag{11}$$

$$\frac{dC_{hi}}{dt} = \frac{\beta_h}{\ell} P_i - \lambda_h C_{hi} \tag{12}$$

where Z is the number of the zones (in this case 14) and ρ_i is the net reactivity of zone i and is defined as

$$\rho_i = \frac{k_{\infty i} - 1}{k_{\infty i}}, \tag{13}$$

and $k_{\infty i}$ is the infinite medium multiplication factor for zone i.

4.1. Steady State Solution

When the reactor is critical and at equilibrium, the derivatives on the LHS of Equations (11) and (12) will vanish. The equilibrium values of delayed neutron precursors are given by

$$C_{hi,\mathrm{r}} = \frac{\beta_h P_{i,\mathrm{r}}}{\ell \lambda_h}, \tag{14}$$

where, subscript "r" stands for the reference value. Reference condition is when the reactor is operating at 100 %FP. The zonal powers at the steady state condition are solutions to a set of equations in the form of

$$\rho_i P_{i,\mathrm{r}} - \sum_{j=1}^{Z} \alpha_{ij} P_{i,\mathrm{r}} + \sum_{j=1}^{Z} \alpha_{ji} P_{j,\mathrm{r}} = 0. \tag{15}$$

Equation (15) is essentially identical to steady state diffusion equation when it is converted to difference equation form. Finding a physically meaningful solution to this problem results in an eigenvalue problem. Generally, an iterative scheme can be used to solve diffusion equation by applying *source iterations* or *power iterations* method to the equations. However, in case of coarse mesh nodalization of CANDU reactor, because of the small dimension of the problem ($Z = 14$), a direct eigenvalue method can be utilized.

The infinite medium multiplication factor for node i is expressed as

$$k_{\infty i} = \frac{\nu \Sigma_{\mathrm{f}i}}{\Sigma_{\mathrm{a}i}}, \tag{16}$$

and the reactivity can be written as

$$\rho_i = 1 - \frac{\Sigma_{\mathrm{a}i}}{\nu \Sigma_{\mathrm{f}i}}. \tag{17}$$

To determine the critical solution of the reactor there is a systematic procedure which yields only the physical solutions. The procedure is based on the assumption that the number of neutrons emitted per fission (ν) can be changed [12]. Essentially, in this method the average number of neutrons emitted per fission is divided by the effective multiplication factor k, therefore the reactivity of each zone can be written as

$$\rho_i = 1 - \frac{k \Sigma_{\mathrm{a}i}}{\nu \Sigma_{\mathrm{f}i}}. \tag{18}$$

Substituting Equation (18) in Equation (15), and expanding the resultant equation, gives

$$P_{i,\mathrm{r}} - \sum_{j=1}^{Z} \alpha_{ij} P_{i,\mathrm{r}} + \sum_{j=1}^{Z} \alpha_{ji} P_{j,\mathrm{r}} = \frac{k \Sigma_{\mathrm{a}i}}{\nu \Sigma_{\mathrm{f}i}} P_{i,\mathrm{r}}. \tag{19}$$

Writing all the equations for $i = 1, \ldots, Z$, gives a clue as how to write the equations for all zones in a compact matrix form. Defining the power vector as $\boldsymbol{P}_{\mathrm{r}} = \begin{bmatrix} P_{1,\mathrm{r}} & P_{2,\mathrm{r}} & \ldots & P_{Z,\mathrm{r}} \end{bmatrix}^{\mathrm{T}}$ one can write

$$\boldsymbol{P}_{\mathrm{r}} + A^{\mathrm{T}} - N\boldsymbol{P}_{\mathrm{r}} = \frac{k}{\nu} M \boldsymbol{P}_{\mathrm{r}}, \tag{20}$$

where A is the coupling coefficients matrix with entries α_{ij} for $i \neq j$ and 0 for $i = j$ (the main diagonal of the matrix are all zeros). M and N are both diagonal matrices defined as

$$M = \text{diag}\left(\frac{\Sigma_{\text{a}i}}{\Sigma_{\text{f}i}}, i = 1, \ldots, Z\right) \tag{21}$$

$$N = \text{diag}\left(\sum_{j=1}^{Z} \alpha_{ij}, i = 1, \ldots, Z\right). \tag{22}$$

Since M is nonsingular, it is easy to calculate its inverse. Subsequently, Equation (20) can be rearranged into

$$M^{-1}\left(I_Z + A_{\text{T}} - N\right)\boldsymbol{P}_{\text{r}} = \frac{k}{\nu}\boldsymbol{P}_{\text{r}}, \tag{23}$$

where, I_Z is the identity matrix of dimension Z. Equation (23) is in standard eigenvalue problem form. The eighenvalues of the matrix $M^{-1}\left(I_Z + A^{\text{T}} - N\right)$ will be equal to k/ν. Alternatively, it can be said that k is the eigenvalue of the following eigenvalue problem:

$$K\left(I_Z + A^{\text{T}} - N\right)\boldsymbol{P}_{\text{r}} = k\boldsymbol{P}_{\text{r}}, \tag{24}$$

where $K = \text{diag}\left(k_{\infty i}, i = 1, \ldots, Z\right)$. The steady state zonal powers are the eigenvectors associated with Equation (23) or Equation (24).

Generally, a matrix of dimension Z has Z eigenvalues and eigenvectors. Next question that arises is which one of these eigenvalue/eigenvector pairs should be chosen as the physically meaningful solution as the reactor steady state zonal powers. This question has been studied in the context of applied mathematics for multigroup diffusion equation and the results can be extended to this simplified case too. It has been proved that all the eigenvalues of steady state multigroup diffusion equation are real, and the largest eigenvalue corresponds to the fundamental harmonic of the flux [13]. Therefore, one only needs to calculate the largest eigenvalue of the matrices on the LHS of the Equation (23) or Equation (24) and its corresponding eigenvector. It should be emphasized that the eigenvector that is calculated, should be multiplied by a scaler to give the real power of the zones. If P_{bulk} is the bulk power of the reactor, and $\boldsymbol{\nu}$ is the eigenvector of Equation (24), the steady state zonal powers are given as

$$\boldsymbol{P}_{\text{r}} = \frac{P_{\text{bulk}}}{\sum_{j=1}^{Z} \nu_j}\boldsymbol{\nu}. \tag{25}$$

If the homogenized averaged absorption and fission cross sections in all zones are equal, i.e., $\Sigma_{\text{a}i} = \Sigma_{\text{a}j}$ and $\Sigma_{\text{f}i} = \Sigma_{\text{f}j}$ for all $i \neq j$, the eigenvalue calculation can be skipped based on a simple physical fact. In such a configuration, the power of each zone is dependent on the volume of the zone. If the volume of the zones is arranged in a vector $\boldsymbol{V} = \begin{bmatrix} V_1 & V_2 & \ldots & V_Z \end{bmatrix}^{\text{T}}$, it can be said $\boldsymbol{\nu} \propto \boldsymbol{V}$ and eigenvalue/eigenvector calculation need not be performed. MATLAB gives the eigenvectors in a normalized form so that the Euclidean length of the vector is equal to one [14]. Therefore, the eigenvector $\boldsymbol{\nu}$ calculated by MATLAB is

$$\boldsymbol{\nu} = \frac{\boldsymbol{V}}{\|\boldsymbol{V}\|_2}. \tag{26}$$

4.2. Nondimensionalization of the Model Equations

Since all the detectors and sensors read the power (flux) and other variables in terms of fractions of the maximum or nominal value of that parameter, it is a good idea to write the dynamic equations

that complies with this practice. To achieve this goal, Equations (11) and (12) can be written in nondimensionalized form. The following variables are defined

$$\hat{P}_i = \frac{P_i}{P_{i,\mathrm{r}}} \tag{27}$$

$$\hat{C}_{hi} = \frac{C_{hi}}{C_{hi,\mathrm{r}}}\,. \tag{28}$$

Replacing these variables in Equations (11) and (12) and simplifying the equations, the nondimensionalized form of the reactor kinetics are obtained:

$$\ell \frac{\mathrm{d}\hat{P}_i}{\mathrm{d}t} = \left(\rho_i - \beta\right)\hat{P}_i - \sum_{j=1}^{Z} \alpha_{ij}\hat{P}_i + \sum_{j=1}^{Z} \eta_{ji}\hat{P}_j + \sum_{h=1}^{m_d} \beta_h \hat{C}_{hi} \tag{29}$$

$$\frac{\mathrm{d}\hat{C}_{hi}}{\mathrm{d}t} = \lambda_h \left(\hat{P}_i - \hat{C}_{hi}\right) \tag{30}$$

where η_{ij} are the elements of a modified coupling coefficients matrix and they are given as

$$\eta_{ij} \triangleq \alpha_{ij} \frac{P_{i,\mathrm{r}}}{P_{j,\mathrm{r}}}\,. \tag{31}$$

The matrix H is built from elements η_{ij} with similar structure to A.

4.3. Implementation of the Reactor Model

If the equations for all zones are written in a matrix notation, they can be implemented in SIMULINK in a vectorized form. For example to implement the power dynamics equation, one does not need 14 integrators. One integrator can be used to implement the vector form of the equation and MATLAB/SIMULINK will expand the equations to the appropriate size.

The nondimensionalized form of the power dynamics for all zones is

$$\ell \frac{\mathrm{d}\hat{\boldsymbol{P}}}{\mathrm{d}t} = R\hat{\boldsymbol{P}} - N\hat{\boldsymbol{P}} + H^{\mathrm{T}}\hat{\boldsymbol{P}} + B\hat{\boldsymbol{C}}\,, \tag{32}$$

where, R is a diagonal matrix defined as

$$R = \mathrm{diag}\left(\rho_i - \beta, i = 1, \ldots, Z\right), \tag{33}$$

and B is a block-diagonal matrix

$$B = \begin{bmatrix} \beta_1 & \beta_2 & \cdots & \beta_{m_d} & : & & & & : & & & & \\ & & & & : & \beta_1 & \beta_2 & \cdots & \beta_{m_d} & : & & & \\ & & & & : & & & & : & \ddots & & & \\ & & & & : & & & & : & & \beta_1 & \beta_2 & \cdots & \beta_{m_d} \end{bmatrix}. \tag{34}$$

The vector $\hat{\boldsymbol{C}}$ is a column vector of dimension $m_d Z \times 1$ of all the delayed neutron precursor concentrations in all nodes for all groups of delayed neutrons and photo-neutrons, i.e.,

$$\hat{\boldsymbol{C}} = \begin{bmatrix} \hat{C}_{11} & \hat{C}_{21} & \cdots & \hat{C}_{m_d 1} & \vdots & \hat{C}_{12} & \hat{C}_{22} & \cdots & \hat{C}_{m_d 2} & \vdots & \cdots & \vdots & \hat{C}_{1Z} & \hat{C}_{2Z} & \cdots & \hat{C}_{m_d Z} \end{bmatrix}^{\mathrm{T}} \tag{35}$$

Similarly, the delayed neutron precursor dynamics for all zones and all groups can be put into vector form

$$\frac{\mathrm{d}\hat{\boldsymbol{C}}}{\mathrm{d}t} = -L_1\hat{\boldsymbol{C}} + L_2\hat{\boldsymbol{P}}\,, \tag{36}$$

where L_1 is an $m_d Z \times m_d Z$ diagonal block-matrix, with its diagonal elements being diagonal matrices themselves. If Λ_1 is defined as

$$\Lambda_1 = \mathrm{diag}\left(\lambda_1, \lambda_2, \ldots, \lambda_{m_d}\right), \tag{37}$$

the block-matrix L_1 can be written as

$$L_1 = \mathrm{diag}\underbrace{\left(\Lambda_1, \Lambda_1, \ldots, \Lambda_1\right)}_{Z}. \tag{38}$$

Matrix L_2 is similar in structure to B, but each one of the diagonal blocks is a column vector.

$$L_2 = \begin{bmatrix} \lambda_1 & \lambda_2 & \cdots & \lambda_{m_d} & : & & & & : & & & & \\ & & & & : & \lambda_1 & \lambda_2 & \cdots & \lambda_{m_d} & : & & & \\ & & & & : & & & & : & \ddots & & & \\ & & & & : & & & & : & & \lambda_1 & \lambda_2 & \cdots & \lambda_{m_d} \end{bmatrix}^{\mathrm{T}}. \tag{39}$$

With the definition of vectors and matrix gains completed, the model can be implemented in Simulink easily. To covert a vector to a diagonal matrix with the elements of the vector on the main diagonal, Matlab's `diag` command is used. To construct the block-diagonal matrices the `blkdiag` is employed. For more information on the calling format of these commands, the interested reader can refer to Matlab help files.

Matlab performs matrix and vector multiplications in several ways (element-wise product, matrix product, etc.). The desired multiplication method should be chosen to get the correct results. The selection of multiplication method is performed within the dialog window of the gain block.

The implementation is shown in Figure 2. This implementation can be compared to the previous implementation of the reactor in [2]. In fact, the current implementation of the model is much simpler and the connections between the elements are better seen. Another benefit of the vectorization is the fact that the older model, includes 14 subsystems, each for one zone of the reactor, and only 1 group of effective delayed neutron precursors are considered in the model. On the contrary to that idea, depending on the size of $\boldsymbol{\beta}$ and $\boldsymbol{\lambda}$, the delayed neutron precursor dynamics is expanded to include 1, 6, 15, or 17 groups of delayed neutron precursors and photo-neutron groups.

Figure 2 shows the implementation of the reactor model with the assumption that there are 14 zones and 17 groups of delayed neutron precursors (6 groups of delayed neutrons, and 11 groups of photo-neutrons). The dimension of each line is shown on the diagram. Most of the blocks on the diagram don't need any explanation. To make it easier to understand the connection of the diagram and Equations (32) and (36), some of the blocks need to be explained briefly.

On the left hand side of the diagram the constant blocks designated by `alpha` and `eta'` are the A and H^{T} matrices, respectively. The "Matlab Function" block `f1` applies the `diag` function on its input to generate a 14×14 matrix. The `f2` Matlab Function block creates the N matrix using the `diag` and `sum` functions.

The `g1` gain block is the matrix gain B and `g3` is the implementation of $\left(R - N + H^{\mathrm{T}}\right)\hat{\boldsymbol{P}}$ in Equation (32). The gain blocks `g2` and `g4` are the representations of $-L_1$ and L_2, respectively. The type of each gain block is shown on its icon. For example `K*uvec` means that a matrix gain `K` is pre-multiplied to the vector `uvec`.

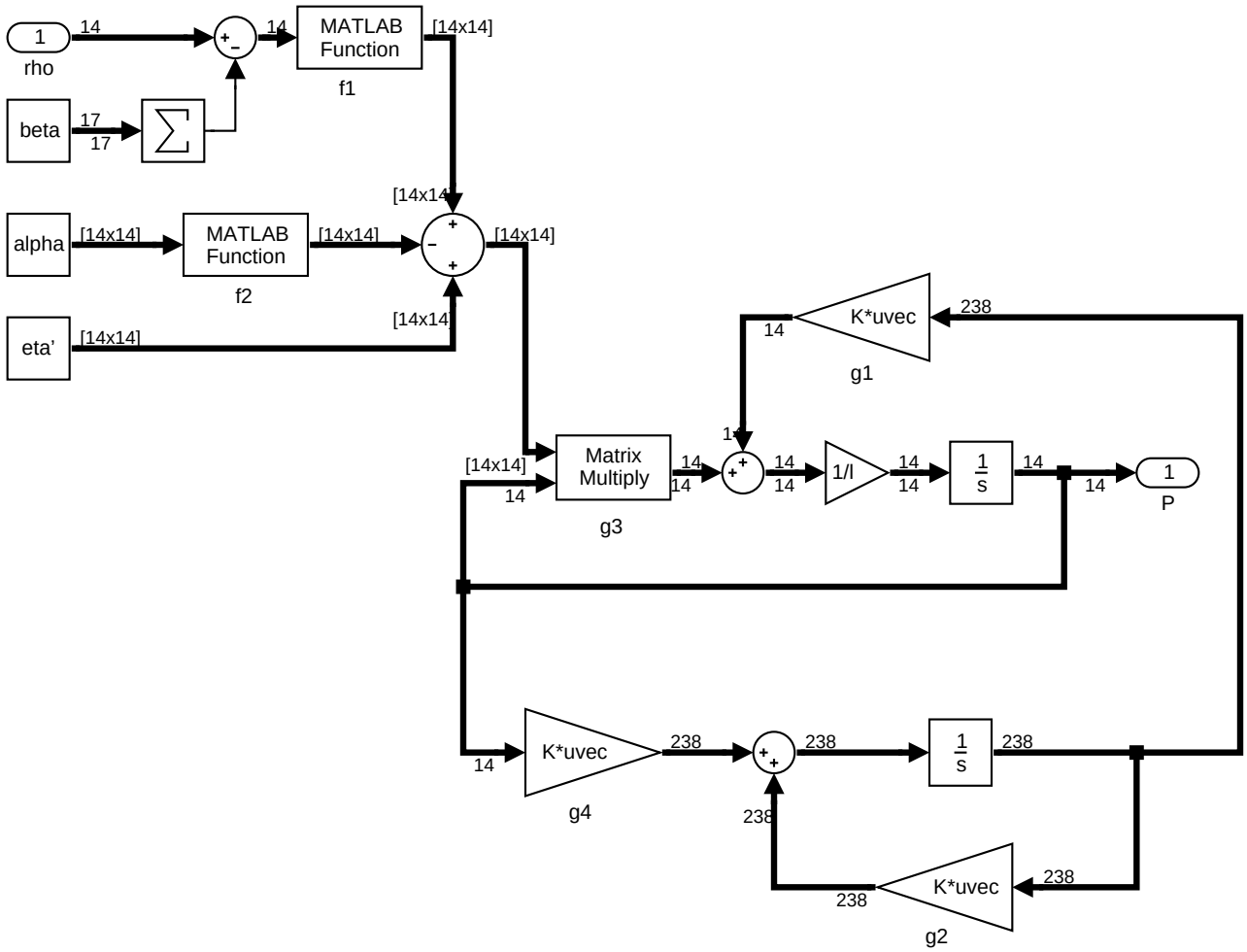

Figure 2: Implementation of the CANDU reactor model in SIMULINK

One advantage of this implementation over the previous one is that it can be used for different types of reactors with different number of zones (nodes), provided that the coupling coefficients matrix is supplied correctly to the model. The model is very flexible to adapt itself to different number of nodes and different number of delayed neutron groups.

5. Xenon Feedback

The iodine and xenon dynamics in each zone can be described by

$$\frac{\mathrm{d}\tilde{I}_i}{\mathrm{d}t} = \gamma_{\mathrm{I}}\Sigma_{\mathrm{f}i}\Phi_i - \lambda_{\mathrm{I}}\tilde{I}_i \tag{40}$$

$$\frac{\mathrm{d}\tilde{X}_i}{\mathrm{d}t} = \gamma_{\mathrm{X}}\Sigma_{\mathrm{f}i}\Phi_i + \lambda_{\mathrm{I}}\tilde{I}_i - \left(\lambda_{\mathrm{X}} + \sigma_{\mathrm{aX}}\Phi_i\right)\tilde{X}_i\,, \tag{41}$$

where $\tilde{I}_i$ and $\tilde{X}_i$ are the number densities of iodine and xenon in m^{-3}. Substituting Φ_i with P_i using Equation (9), Equations (40) and (41) can be converted to

$$\frac{\mathrm{d}I_i}{\mathrm{d}t} = \gamma_\mathrm{I}\Sigma_{\mathrm{f}i}P_i - \lambda_\mathrm{I}I_i \tag{42}$$

$$\frac{\mathrm{d}X_i}{\mathrm{d}t} = \gamma_\mathrm{X}\Sigma_{\mathrm{f}i}P_i + \lambda_\mathrm{I}I_i - \left(\lambda_\mathrm{X} + \bar{\sigma}_{\mathrm{X}i}P_i\right)X_i\,, \tag{43}$$

where the new variables I_i and X_i are defined as $I_i = E_\mathrm{f}\Sigma_{\mathrm{f}i}V_i\tilde{I}_i$ and $X_i = E_\mathrm{f}\Sigma_{\mathrm{f}i}V_i\tilde{X}_i$. The xenon absorption cross section for each zone is scaled according to

$$\bar{\sigma}_{\mathrm{X}i} = \frac{\sigma_\mathrm{aX}}{E_\mathrm{f}\Sigma_{\mathrm{f}i}V_i}\,. \tag{44}$$

It is important to note that $I_i(t)$ and $X_i(t)$ are measured in J/m and $\bar{\sigma}_{\mathrm{X}i}$ is measured in J^{-1}.

5.1. Steady State Solution

At reference conditions, steady state values of iodine and xenon are given by

$$I_{i,\mathrm{r}} = \frac{\gamma_\mathrm{I}\Sigma_{\mathrm{f}i}P_{i,\mathrm{r}}}{\lambda_\mathrm{I}} \tag{45}$$

$$X_{i,\mathrm{r}} = \frac{\left(\gamma_\mathrm{I}+\gamma_\mathrm{X}\right)\Sigma_{\mathrm{f}i}P_{i,\mathrm{r}}}{\lambda_\mathrm{X} + \bar{\sigma}_{\mathrm{X}i}P_{i,\mathrm{r}}}\,. \tag{46}$$

As can be seen, the steady state value of xenon concentration is a nonlinear function of the steady state power of the zone.

5.2. Nondimensionalization of Xenon Model

It is desirable to nondimensionalize the iodine and xenon dynamics equations, similar to power and delayed neutron dynamics equations. To achieve this, the power of each zone is nondimensionalized according to Equation (29), and the iodine and xenon concentrations in nondimensionalized frame are $\hat{I}_i = I_i \big/ I_{i,\mathrm{r}}$ and $\hat{X}_i = X_i \big/ X_{i,\mathrm{r}}$. Replacing the I_i and X_i with their nondimensionalized representations, the iodine and xenon dynamics can be written as

$$\frac{\mathrm{d}\hat{I}_i}{\mathrm{d}t} = \lambda_\mathrm{I}\left(\hat{P}_i - \hat{I}_i\right) \tag{47}$$

$$\frac{\mathrm{d}\hat{X}_i}{\mathrm{d}t} = \frac{\lambda_\mathrm{X} + \bar{\sigma}_{\mathrm{X}i}P_{i,\mathrm{r}}}{\gamma_\mathrm{I}+\gamma_\mathrm{X}}\left(\gamma_\mathrm{X}\hat{P}_i + \gamma_\mathrm{I}\hat{I}_i\right) - \left(\lambda_\mathrm{X} + \bar{\sigma}_{\mathrm{X}i}P_{i,\mathrm{r}}\hat{P}_i\right)\hat{X}_i\,. \tag{48}$$

As it is seen from the above equations, the steady state power of the zone $P_{i,\mathrm{r}}$ appears as a parameter in the xenon equation in nondimensionalized form.

Further simplification can be made with the assumption that the fission cross section of all zones are equal. As previously shown, in such a case, the steady state powers of the reactor can be expressed in terms of the zone volumes as described in Equation (26). Steady state power of zones can be written as $\boldsymbol{P}_\mathrm{r} = \varepsilon\boldsymbol{V}$ where $\varepsilon = P_\mathrm{bulk} \Big/ \sum_{j=1}^{Z} V_j$ and the product of $\bar{\sigma}_{\mathrm{X}i}P_{i,\mathrm{r}}$ can be simplified as

$$\bar{\sigma}_{\mathrm{X}i}P_{i,\mathrm{r}} = \frac{\sigma_\mathrm{aX}}{E_\mathrm{f}\Sigma_\mathrm{f}V_i} \times \varepsilon V_i = \frac{\varepsilon\sigma_\mathrm{aX}}{E_\mathrm{f}\Sigma_\mathrm{f}} \triangleq c_\mathrm{X}\,. \tag{49}$$

This new parameter c_X can be considered as the xenon burnout parameter, and it is not dependent to the zoning of the reactor geometry. Since the available data for CANDU reactor gives equal averaged fission cross section for all zones, this c_X is used as the burnout parameter.

Substituting c_X for $\bar{\sigma}_{Xi}P_{i,r}$ in Equation (48) results in the xenon dynamics equation of zone i for the equal fission cross sections

$$\frac{d\hat{X}_i}{dt} = \frac{\lambda_X + c_X}{\gamma_I + \gamma_X}\left(\gamma_X\hat{P}_i + \gamma_I\hat{I}_i\right) - \left(\lambda_X + c_X\hat{P}_i\right)\hat{X}_i\,. \tag{50}$$

5.3. Implementation of Xenon Feedback Model

Iodine and xenon dynamics equation for all zones can be written in a vector form. Iodine dynamics in vector is written as

$$\frac{d\hat{\boldsymbol{I}}}{dt} = \lambda_I\left(-\hat{\boldsymbol{I}} + \hat{\boldsymbol{P}}\right)\,. \tag{51}$$

The nonlinear term in xenon dynamics makes it more difficult to put the equations for all zones into vector form. To achieve this goal, one has to define an element-wise vector product symbol. MATLAB performs the element-wise product using the `.*` binary operator. The $\odot$ symbol is used to represent the element-wise product in the xenon dynamics equation. Using this convention, xenon dynamics in vector form is given as

$$\frac{d\hat{\boldsymbol{X}}}{dt} = \frac{\lambda_X + c_X}{\gamma_I + \gamma_X}\left(\gamma_X\hat{\boldsymbol{P}} + \gamma_I\hat{\boldsymbol{I}}\right) - \lambda_X\hat{\boldsymbol{X}} - c_X\hat{\boldsymbol{P}}\odot\hat{\boldsymbol{X}}\,. \tag{52}$$

The xenon load reactivity for each zone is defined as

$$\rho_{X,i} = -\frac{\bar{\sigma}_{Xi}X_i}{\Sigma_{ai}}\,. \tag{53}$$

If the nondimensionalized value of X_i is used in Equation (53), and simplifications are made to the resulting equation, the xenon load reactivity in terms of $\hat{X}_i$ can be expressed as

$$\rho_{X,i} = -\frac{\Sigma_{fi}}{\Sigma_{ai}}\frac{\left(\gamma_I + \gamma_X\right)\bar{\sigma}_{Xi}P_{i,r}}{\lambda_X + \bar{\sigma}_{Xi}P_{i,r}}\hat{X}_i\,. \tag{54}$$

Figure 3 depicts the SIMULINK implementation of the iodine and xenon dynamics. The input to the model is nondimensionalized reactor power, and the output of the model is the xenon load reactivity in k. It is assumed that the fission and absorption cross sections of all zones are equal, therefore the c_X formulation is used in the implementation.

The top integrator in Figure 3 is the integrator for all the iodine concentrations in 14 zones. The bottom integrator integrators the xenon dynamics equations in 14 zones. The gains `g1-g6` in the block diagram represent the coefficients in Equations (51–52). `g1` is the scaler λ_I, `g2` is the scaler $\gamma_X\left(\lambda_X + c_X\right)/\left(\gamma_I + \gamma_X\right)$, `g3` is scaler $\gamma_I\left(\lambda_X + c_X\right)/\left(\gamma_I + \gamma_X\right)$, `g4` and `g5`, are simply c_X and λ_X, respectively. `g6` is the coefficient relating xenon load to xenon concentration in Equation (54). The multiplier in the block diagram performs element-wise product on vectors $\hat{\boldsymbol{P}}$ and $\hat{\boldsymbol{X}}$ using MATLAB's `.*` operator.

6. Conclusion

Mathematical models of CANDU reactor and xenon feedback dynamics are developed in this paper. The models are implemented in MATLAB/SIMULINK are parts of a unified CANDU control system simulation platform. The models of other elements of flux control loop are developed. It is concluded that the MATLAB/SIMULINK provide an excellent simulation environment for studying reactor dynamics. The vectorization of the models is particularly useful.

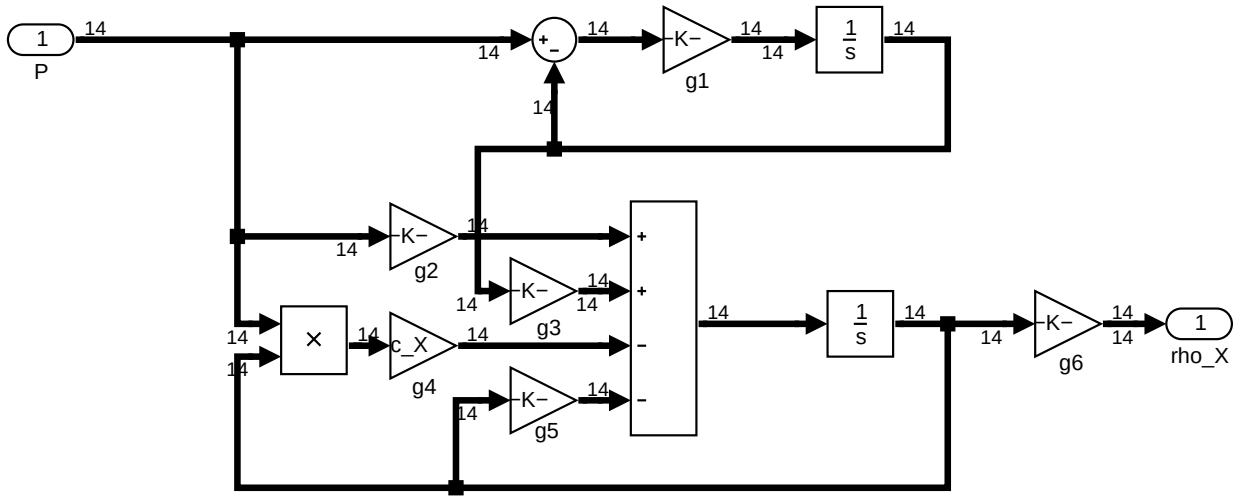

Figure 3: Implementation of iodine, xenon and xenon load reactivity in SIMULINK

Acknowledgement

The authors would like to acknowledge the financial support from NSERC and UNENE as well as the technical supports from the UNENE industrial partners for the work reported in this paper.

References

[1] K. Lam and M. J. MacBeth, "Functional model for simulator based training in the pacific basin," in *Proceedings of 11th Pacific Basin Nuclear Conference*, vol. 2. Banff, AB, Canada: Canadian Nuclear Society, May 1998, pp. 1349–1356. [Online]. Available: http://www.cti-simulation.com/Simulation_articles.htm#functional_model_for_Simulator_Based_Training_in_the_Pacific_Basin_

[2] H. Javidnia and J. Jiang, "MATLAB/SIMULINK model of CANDU reactor for control studies," in *Proceedings of the 27th Annual CNS conference*. Toronto: Canadian Nuclear Society, 2006.

[3] A. P. Tiwari, "Modeling and control of a large pressurized heavy water reactor," Ph.D. dissertation, Indian Institute of Technology, Bombay, 1999. [Online]. Available: http://www.sc.iitb.ac.in/theses/phd_theses/tiwari_thesis.pdf

[4] G. Sharma, B. Bandyopadhyay, and A. Tiwari, "Spatial control of a large pressurized heavy water reactor by fast output sampling technique," *IEEE Transactions on Nuclear Science*, vol. 50, no. 5 III, pp. 1740–1751, 2003. [Online]. Available: http://dx.doi.org/10.1109/TNS.2003.818271

[5] G. I. Bell and S. Glasstone, *Nuclear Reactor Theory*. New York: Van Nostrand Reinhold Company, 1970.

[6] J. J. Duderstadt and L. J. Hamilton, *Nuclear Reactor Analysis*. John Wiley and Sons, Inc., 1976.

[7] B. Arsenault, "General introduction to methods of physics analysis and physics computer codes," AECL, Canada, Workshop Notes, March 2000. [Online]. Available: http://canteach.candu.org/library/20054302.pdf

[8] ——, "RFSP model," AECL, Canada, Workshop Notes, March 2000. [Online]. Available: http://canteach.candu.org/library/20054304.pdf

[9] A. Hoeld and O. Lupas, "Real-time simulation of the transient behavior of local and global pressurized water reactor core and plant parameters," *Nuclear Science and Engineering*, vol. 85, no. 4, pp. 396–417, December 1983.

[10] M. V. Gregory, B. N. Aviles, and S. J. Yakura, "A three-dimensional neutronics model for reactor training simulators," *Nuclear Science and Engineering*, vol. 92, no. 3, pp. 372–381, March 1986.

[11] W. M. Stacey, *Space-time Nuclear Reactor Kinetics*, ser. Nuclear Science and Technology. New York: Academic Press, 1969, vol. 5.

[12] A. F. Henry, *Nuclear-Reactor Analysis*. Cambridge, Massachusettes: The MIT Press, 1975.

[13] G. J. Habetler and M. A. Martino, "Existing theorems and spectral theory for the multigroup diffusion model," in *Proceedings of the Eleventh Symposium in Applied Mathematics*, G. Birkhoff and E. Wigner, Eds., vol. XI, Nuclear Reactor Theory. Providence, RI: American Mathematical Society, 1961, pp. 127–139.

[14] MATLAB *7: Mathematics*. Natick, MA: The Mathworks, 2007. [Online]. Available: http://www.mathworks.com/access/helpdesk/help/pdf_doc/matlab/math.pdf