TRANSPORT-DIFFUSION COUPLING FOR CANDU REACTOR CORE FOLLOW-UP

Elisabeth Varin, Guy Marleau and Richard Chambon,
Ecole Polytechnique de Montréal, IGN,
P.O. Box 6079, Station Centre-Ville, Montréal H3C 3A7, Canada
elisabeth.varin@polymtl.ca, guy.marleau@polymtl.ca, richard-pierre.chambon@polymtl.ca

Abstract

We couple the finite reactor diffusion code DONJON and the lattice code DRAGON,called for simplicity D&D, to perform reactor follow-up calculations using a history-based approach. In order to do this, a new D&D module is developed. This module manages the transfer of information between standard DONJON and DRAGON data structures. Moreover, it stores in a history data structure the global and local parameters required for cell calculations as well as the isotopic composition of the various materials present in each cell of the reactor.

We then implement in D&D a parallel algorithm to perform history-based CANDU reactor calculations. Here, we assign to each processor a specific number of fuel channels to be analyzed. The DRAGON cell calculations for each of the fuel bundles associated with the specified channels are performed on the same processor in order to minimize communication time. Only the macroscopic cross section libraries are exchanged between the processor. Since the amount of data exchanged is relatively small, we expect to obtain an ideal speed-up.

The coupling is tested for the analysis of a simplified CANDU reactor model with 4×4 channels each containing 4 bundles. A 100 full-power days core tracking sequence with 16 refueling steps is studied. Results are coherent with those obtained using more approximate approaches. Parallel speed-up is near optimal indicating that the use of this approach for more realistic reactor calculations should be pursued.

Introduction

CANDU reactors compensate for reactivity loss by on-power refueling. For CANDU-6 cores, each refueling involves the insertion of 8 new bundles and discharging 8 out of the 12 bundles contained in each channel. The remaining 4 bundles are shifted along the channel. This strategy implies that the fuel management codes used for CANDU core analysis have to treat explicitly each bundle using few-group nuclear cross sections as a function of its burnup — exposure time and specific power — as well as its local parameters. However because a bundle can be moved during refueling from a region of relatively low- (or high-) flux level to one of much higher (or lower) flux, it is difficult to assume that the fuel properties are only dependent of burnup with fixed local parameters, irrespective of their history. This strategy also implies that refueling sites must be selected on a daily basis. The selection process requires the knowledge of the actual bundle burnup and core power distributions as well as controller positions. Because fuel burnup cannot be directly measured, it must therefore be estimated by simulation methods.

Different approximate approaches have been used to treat the local parameters and the history in the core and to provide accurate cross section libraries corresponding to an instantaneous state of the reactor. The standard technique consists in interpolating inside a multidimensional tabulated database.[1] In practice, one is limited to a few interpolation parameters and this technique makes it difficult to take into account the bundle history effect. A second approach consists in using a feedback model with actinide burnup history where one generates a coefficient database for cross section reconstruction as a function of local parameters and bundle history.[2] These two methods were compared against a CANDU core follow up[3, 4] where the diffusion calculations were driven by the diffusion code DONJON.[5]

A more exact technique, called the history-based methodology, has also been developed by Atomic Energy of Canada Limited.[6] In this approach, a lattice code is used for calculating the actual macroscopic cross sections associated with each fuel bundle. This method has been validated on CANDU core tracking with 3–D diffusion calculations driven by the reactor code RFSP,[7] while different transport codes were tested, including POWDERPUFS-V[8], WIMS-AECL[9] and a simplified approach called Simple-Cell Model.[10]. While being the most precise approach, the history-based technique is not widely used because of the large amount of time required to perform the cell calculations. However, the advances in processor performance and parallel computing, now make this technique a more viable alternative.[11]

The goal of this paper is the coupling of the lattice code DRAGON[12] and the diffusion code DONJON and the implementation within the resulting code of a parallel procedure for a history-based study of CANDU reactors. This capability is tested on a simplified 16-channel CANDU reactor model. Here each channel contains 4 fuel bundles and a one-bundle shift refueling strategy is considered. Based on theses results, possible simplifications in the cross section generation would be proposed, evaluated and associated with error margins.

In the first section, the history-based approach is explained. The coupling strategy between DRAGON and DONJON is described in the second section. The parallel strategy we developed is exposed in the third section. Core follow-up studies for the simplified CANDU reactor are performed in the fourth section. The effects of using a history-based approach are then evaluated in terms of reactor reactivity and channel powers when compared to the standard cross section tabulations approach. Coupling scalability, and parallel speed up and requirements are also analyzed. Finally we conclude.

History-based Local Parameter Approach for CANDU Analyses

CANDU-6 reactor cores are composed of $N_c = 380$ horizontal fuel channels located in pressure tubes and filled with $N_b = 12$ fuel bundles. The heavy-water coolant circulating in the pressure tube has different densities and temperatures at the inlet and outlet of each channel. In fact, one generally assumes that the heavy-water coolant surrounding each fuel bundle has a uniform density d^c and temperature T^c. Similarly, the fuel bundle is assumed to be at a fixed temperature T^f and to provide a particular power P^H. As the core depletes, the fuel composition changes and the bundle conditions are affected.

In CANDU reactors, the fuel is replaced when its reactivity loss is too important. The general refueling strategy consists in inserting N fresh fuel bundles at one end of the channel while removing N depleted bundles at the other

end (fueling with N bundle-shift). This on-power refueling procedure has various effects on the local parameter distribution in the core including:

- The presence of fresh fuel bundles at one end of a channel modifies the bundle power distribution in the refueled channel and over the total reactor.
- Some bundles are pushed through the refueled channel and see power densities and local parameters that are different from those they saw at their previous location in the core. This clearly affects the fuel depletion. This aspect of the on-power refueling is identified as the history effect.
- Refueling also tends to increase the reactivity in the core. This is compensated by adjusting the liquid zone controller levels and by changing the poison (boron) concentration in the heavy-water moderator.

Accordingly, the fuel properties will depend on the bundle burnup-history, on global parameters (such as poison concentration in the moderator) as well as on local parameters (such as coolant temperature).

Let us define the state vector $\vec{L} = (T^f, T^c, d^c, d^m, N^b)$ and assume that it can be used to characterized each fuel bundle. Here the global parameters d^m and N^b respectively represent the moderator purity and boron concentration while the local parameters T^f, T^c and d^c were defined previously. The few-group macroscopic cross sections may be represented as:[2]

$$\Sigma_{yg}(\vec{L}, B, P^H) = \sum_i N^i(\vec{L}, B, P^H)\sigma^i_{yg}(\vec{L}, P^H) \tag{1}$$

where B is the fuel burnup, N^i is the concentration and σ^i is the microscopic cross section of interest for isotope i. The indices y and g are respectively the interaction type (absorption, fission, ...) and the energy group. In this formulation, no particular treatment is considered for the fuel history.

In the usual tabulation model, one first selects a range of global and local parameters that are representative of a CANDU core operating window. For every combination of global and local parameters, cell calculations are performed and burnup dependent macroscopic cross sections tables $\Sigma_{yg}(B)$ are produced. In general, the depletion calculations are performed at various power levels to take into account the fact that the fuel bundles may be located in a high or low flux region. Accordingly, this model often implies a large number of transport calculations. Finally, during the diffusion calculations, the state vector $\vec{L}$ associated with each bundle is selected and interpolation is realized over the field of $\vec{L}$ and fuel burnups.

Note that the simplest version of the tabulation model is that where only burnup tabulation is allowed and the transport calculations are performed using nominal values for $\vec{L}$ that represents the average operational core conditions. This approach has been used and validated for design calculations, and for a time-average representation of the reactor. However this approximation is too crude for operation follow-up where the fuel properties have to be tabulated for every operation condition. Moreover no effort is made to take into account the history effect namely, when a channel is refueled, the burnup power density and local parameters of some bundles are changed. As a result, the concentration N^i in Eq. (1) depends both on the actual values of $\vec{L}$ and B but also on the previous values of $\vec{L^H}$ and power P^H. On the other hand, the microscopic cross sections are only function of the actual local and global parameters. This implies that instead of considering explicitly macroscopic cross section tables, the isotope concentrations and the microscopic cross sections should be tabulated independently.

Using this idea, a feedback model method was developed to couple the DRAGON and DONJON codes.[2] Starting from nominal conditions $\vec{L}_0$, the variations in the macroscopic cross sections are expressed as:

$$\Delta\Sigma_{yg}(\vec{L}, B, P^H) = \Sigma_{yg}(\vec{L}, B, P^H) - \Sigma_{yg}(\vec{L}_0, B, P^H) = \Delta\Sigma_{yg}(\vec{L}, B) + \Delta\Sigma_{yg}(\vec{L}, B, P^H) \tag{2}$$

where the effect on the cross sections of a variation of the state vector $(\vec{L} - \vec{L}_0)$ at a fixed reference power P are separated from the history $(\Delta\Sigma_{yg}(B, P^H))$ effects. The history independent component of the macroscopic cross section variation $\Delta\Sigma_{yg}(\vec{L})$ is simulated using a semi-empirical relation that rely on pre-tabulated feedback coefficients. On the other hand $\Delta\Sigma_{yg}(\vec{L}, B, P^H)$ is approximated using

$$\Delta\Sigma_{yg}(\vec{L}, B, P^H) \approx \sum_i \Delta N^i(\vec{L}, B, P^H)\sigma^i_{yg}(\vec{L}, B) \tag{3}$$

where $\Delta N^i(\vec{L}, B, P^H)$ represents the variations in isotopic concentrations resulting from different burnup history. In the feedback model $\Delta N^i(\vec{L}, B, P^H)$ for the most important isotopes is also approximated using semi-empirical relations. This method has enhanced the capabilities in DONJON of evaluating the fuel cross sections over a wide range of local and global parameters. It gives good results in cases of operation follow-up and for coupled neutronics/thermo-hydraulics calculations.[4, 14]

The history-based (HB) approach consists in using a lattice code to compute the cell average few group cross sections for each fuel bundle and for each simulation time step ($\Delta t = T_f - T_i$). This implies:

1. Defining the average local and global parameters (state vector $\vec{L}$) over the simulation time step as well as the power density P associated with a fuel bundle.
2. Recovering the initial (at time T_i) isotopic content of a fuel bundle.
3. Evaluating the initial multi-region and multi-group macroscopic cross section and solving the transport equation for the initial flux.
4. Solving the depletion equation for the final isotopic content of a fuel bundle assuming a constant flux distribution (spatial and multigroup) over the burnup period.
5. Evaluating the final multi-region and multi-group macroscopic cross section and solving the transport equation for the final flux.
6. Producing the final cell averaged few-group cross sections associated with the fuel bundle.

In the case of a CANDU-6 follow-up, the reactor is post-simulated every 2 to 6 days.[4] During that period, most of the fuel bundles remain at the same location in the core and thereby see only slight changes in $\vec{L}$ and P. On the other hand, for the refueled channels, large changes in $\vec{L}$ and P may be observed for the displaced bundles while the isotopic contents for the new fuel bundles must be extracted from the fresh fuel properties. Finally note that in order to evaluate in a consistent way the average power density for each bundle one should also iterate between transport and diffusion calculations.

DRAGON-DONJON Coupling Methodology for History-Based Reactor Calculations

Here we will perform our history-based reactor calculations using DRAGON for the cell calculations and DONJON for the finite reactor calculations in diffusion theory. In fact we have combined DRAGON and DONJON in a single code, called for simplicity D&D. From the programmers point of view the integration of these two codes is trivial since all the routines in DRAGON and DONJON have different names. However, this is not sufficient to couple DRAGON and DONJON in a single execution. Here, the coupling is simplified due to the fact that both codes were developed using the GAN generalized driver[13] where the information between modules is exchanged using well defined data structures.

For reactor analyses, the standard transport-diffusion coupling is characterized by the calculation in transport theory of few-group macroscopic cross sections, then their transfer to the diffusion calculations; no automatic interaction with the transport code is required after a finite reactor calculation. This means that a time averaged CANDU-6 reactor calculation requires a COMPO data structure created by the `CPO:` module of DRAGON and containing the condensed and homogenized cell cross sections tabulated as a function of burnup. This data structure is read directly by the `CRE:` module of DONJON to produce the macroscopic cross section library (a MACROLIB data structure) required for the diffusion calculations. Similarly, the multi-parameter feedback database generated by DRAGON can be used by the `AFM:` module of DONJON to create the MACROLIB data structure required for reactor follow-up.[4]

For history-based studies, a more complex coupling methodology is required. It involves the transfer of information from DRAGON to DONJON described above and the use of DONJON generated information to modify some of the DRAGON data structures. In fact, some cell properties required for a DRAGON calculation are extracted from the DONJON generated FUEL_MAP data structure. This FUEL_MAP data structure contains the bundle

power required for fuel depletion, information on the channels to be refueled during the next DONJON simulation time step and optionally local parameters (fuel temperature, coolant temperature and density, etc.).[14]

Another observation is that DRAGON, as an independent code, is rarely used to analyze a very large number of independent cells in a single execution. Accordingly, the various data structures required for a complete cell calculation are managed manually. In addition, the execution of DRAGON is mainly controlled by the information provided in the input data stream. We avoid the problem of data structure proliferation by creating a global data structure that accumulates independent cell data structures. The transfer of information from a data structure and the input data stream is accomplished via the use of data variables that are managed by the CLE-2000 toolbox used in DRAGON and DONJON.[13] Finally, because we want to avoid the possibility of introducing new errors in the current DRAGON and DONJON modules, we have developed a new module for D&D that will manage the coupling described above.

The new module created for our history-based analysis is called HST:. This module is first used to extract the information stored in a DONJON FUEL_MAP structure and make it available to DRAGON. It is also used to accumulate in a new type of data structure, the global parameters associated with a collection of cells as well as the local parameters associated with each cell. This data structure, of type HISTORY, also contains a description of the isotopic contents of each mixture in each cell at a specific point in time that have been retrieved from DRAGON DEPLETION data structures. Using this information, the HST: module will be able to generate new DEPLETION data structures as well as to store in local CLE-2000 variables the information (power density, depletion time step, global and local parameters) required for a future DRAGON cell burnup calculations.

Let us now describe the algorithm used in our history-based calculations (see Fig. 1). First the geometry of both the reactor and the fuel cell are defined and analyzed. A FUEL_MAP structure is created for N_c channels by N_b bundles in a reactor geometry limited to the fuel bundles (no reflector information). This structure is then updated with previously computed power distribution. These initial and general fuel core conditions are used to initialize the HISTORY data structure using the HST: module. Here, the location of each fuel bundle in the core is identified. The module HST: associates automatically to each of these bundles both a cell and a fuel type and create, if required, sub-directories in the HISTORY data structure that will contain cell or fuel type information. In this context, a specific fuel type is characterized by its geometry and fresh fuel isotopic composition. On the other hand, a cell type is characterized by the fuel type it contains, the local parameters it sees and its current depletion state.

The full reactor simulation period is generally divided into time steps. In the DONJON code, a simulation time step is first defined by the REFUEL: module and stored in the FUEL_MAP structure. When the effective multiplication factor for the core is below a certain critical value, the refueling procedure is activated. This refueling procedure consists in identifying when and where fresh bundles need to be inserted in the core. In the DONJON code, the selection of the channels to be refueled is also controlled by the REFUEL: module. Typically, during a simulation time step $\Delta T = T_f - T_i$, k fuel channels will be refueled at times t_1 to t_k such that

$$T_i \leq t_1 \leq \cdots \leq t_k \leq T_f.$$

The module REFUEL: stores in the FUEL_MAP data structure, the simulation time step ΔT to be considered as well as $\Delta t_k = t_k - T_i$, the time before a channel is refueled.

Using the refueling information of the FUEL_MAP data structure, the HISTORY data structure is updated prior to depletion calculations in DRAGON (see Fig. 1). For the cells located in channels where no refueling takes place, this simply means updating the local parameters and power densities. For the cells located in a channel refueled at time t_k, the process is somewhat more complex. In the case where N_r out of the N_b bundles in a channel are refueled, the following situations will occur:

1. The bundles at locations $N_b - N_r + 1$ to N_b are extracted.
2. The bundles at locations 1 to $N_b - N_r$ are displaced to location $N_r + 1$ to N_b.
3. Fresh bundles are inserted at location 1 to $N_b - N_r$.

Accordingly, the HST: module first eliminates from the HISTORY data structure all the bundles to be extracted from the core during the time step ΔT. For the displaced bundles, the HST: module creates two sets of local properties,

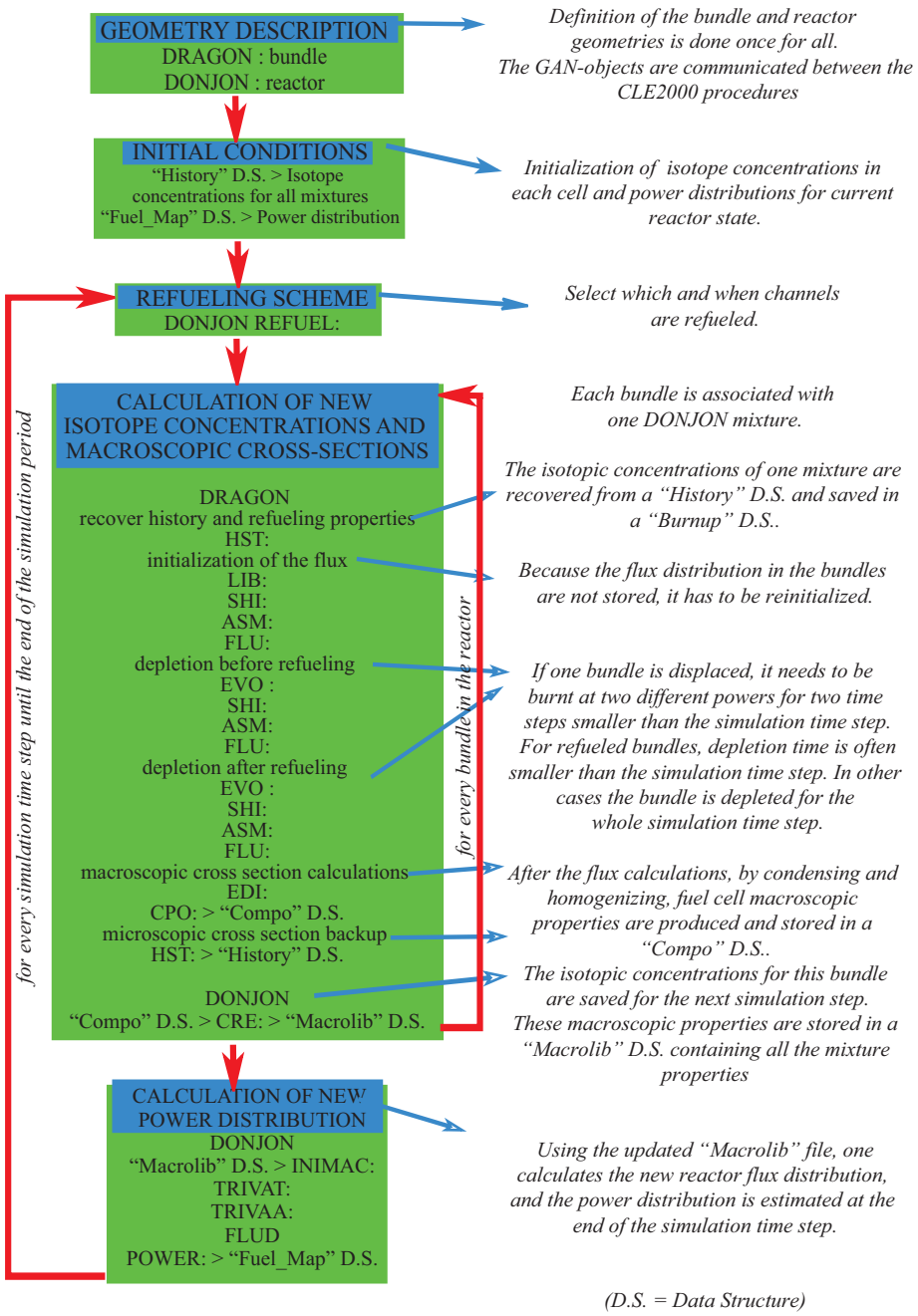

Figure 1: Sequential Algorithm for History-Based D&D Calculations

power density and depletion time records, the first set corresponding to the properties before refueling (location 1 to $N_b - N_r$) and the second set to the properties after refueling (location $N_r + 1$ to N_b). Finally, for the fresh bundles, a single set of local properties, power density and depletion time records is created corresponding to the properties after refueling (location 1 to $N_b - N_r$). In addition, the initial isotopic composition of fresh fuel bundles is recovered from the fuel type directory associated with the refueled cell.

Once the HISTORY data structure has been updated, the `HST:` module is used to extract from it the information required by DRAGON for each cell calculation for the next simulation time step ΔT. The isotopic composition

of the cell at T_i is stored in a new DEPLETION data structure. Two sets of burnup and local parameters are also generated and stored in local CLE-2000 variables, these sets being associated with the properties of the cell before and after refueling. Note that in the case of fresh fuel bundles, the parameters associated with the before refueling set should not be used as indicated by the fact that the depletion time step in this case vanishes. One then proceeds with the cell calculation, which will update the current DEPLETION data structure until time T_f is reached. At this point the HISTORY data structure is again modified to take into account the new isotopic concentrations of each cell. A COMPO data structure associated with the final isotopic composition and flux distribution is generated and accumulated in a MACROLIB data structure. Once all the cells have been analyzed, a finite reactor calculation in diffusion theory is performed using the global reactor macroscopic cross section database. The resulting power distribution in the core is then stored in a FUEL_MAP data structure. This process is repeated until the complete simulation period has been covered.

Parallel Strategy

Most of the CPU time in a coupled DRAGON/DONJON calculation is spent in the expensive and repetitive cell calculations. For a reactor containing N_b fuel bundles in each of its N_c channels, $N_c \times N_b$ cell calculations are required at each time step between reactor calculations. Because the cells are assumed independent, the parallel strategy that we selected consists in distributing the cell calculations according to the channel number on N_p processors (PE). The number of fuel channels distributed on each processor depends on its speed in order to minimize the calculation time, between refueling, thereby obtaining an acceptable speedup.

Since the information required for these cell calculations or retrieved from them is in the form of data structures, the data exchanged between the processors is limited to a minimum.[13] Parallel modules[15] previously developed with PVM were adapted to MPI[16] routines. The new module `DRVMPI:` developed for D&D manage the parallel execution. It defines the communication group, it returns to the main input stream the total number of PE selected as well as the explicit process number associated with each PE through CLE-2000 data variables. It also sends a barrier signal to all the processes and computes the repartition limits on each process to respect the load balancing requirements. In order to exchange data structures, the `SNDMPI:` module was also programmed. This general-purpose module can send or receive DRAGON and DONJON data structure. Using these MPI based modules, the history-based algorithm was modified as shown in Figure 2.

Since the finite reactor calculations consume a very small part of the global CPU time, they are realized on every PE. This minimizes the process idle time and the communication load. After defining the simulation time step and the refueled channels using the `REFUEL:` module, distributed cell calculations are realized on every processor using the FUEL_MAP information. A HISTORY structure is maintained on every PE. It contains the bundle information, power density, local parameters for the whole core, but only the isotopic concentrations for the bundles analyzed on this process. This results in a distributed storage of the reactor HISTORY data structure.

On each PE, the macroscopic cross sections are stored for the required channels only. The full reactor MACROLIB structure is the concatenation of these distributed MACROLIB. This operation is performed using the `SNDMPI:` module and the `DMAC:` module of DONJON. The latter can be used to add or substract MACROLIB structures and was initially programmed for kinetics calculations. As shown in Figure 2, the addition of MACROLIB on a single processor is obtained by a succession of send/receive and barrier operations. Finally the reactor MACROLIB structure is broadcast on every PE, where it will be used for the next reactor calculation. Since the number of communication is N_p MACROLIB send/receive, we expect that the speed-up will be almost ideal.

Results

A simplified model of a 3–D reactor with 16 channels each containing 4 bundles is used to test our coupling approach. In order to start these history-based calculations, an instantaneous reactor model must first be defined. To generate such a model, we use an approach based on the channel age model found in CANDU fuel management design codes where a specific burnup is associated with each bundle reflecting a particular channel refueling history.[7] Accordingly, a time average calculation is first carried out on the simplified model. There, the macroscopic cross

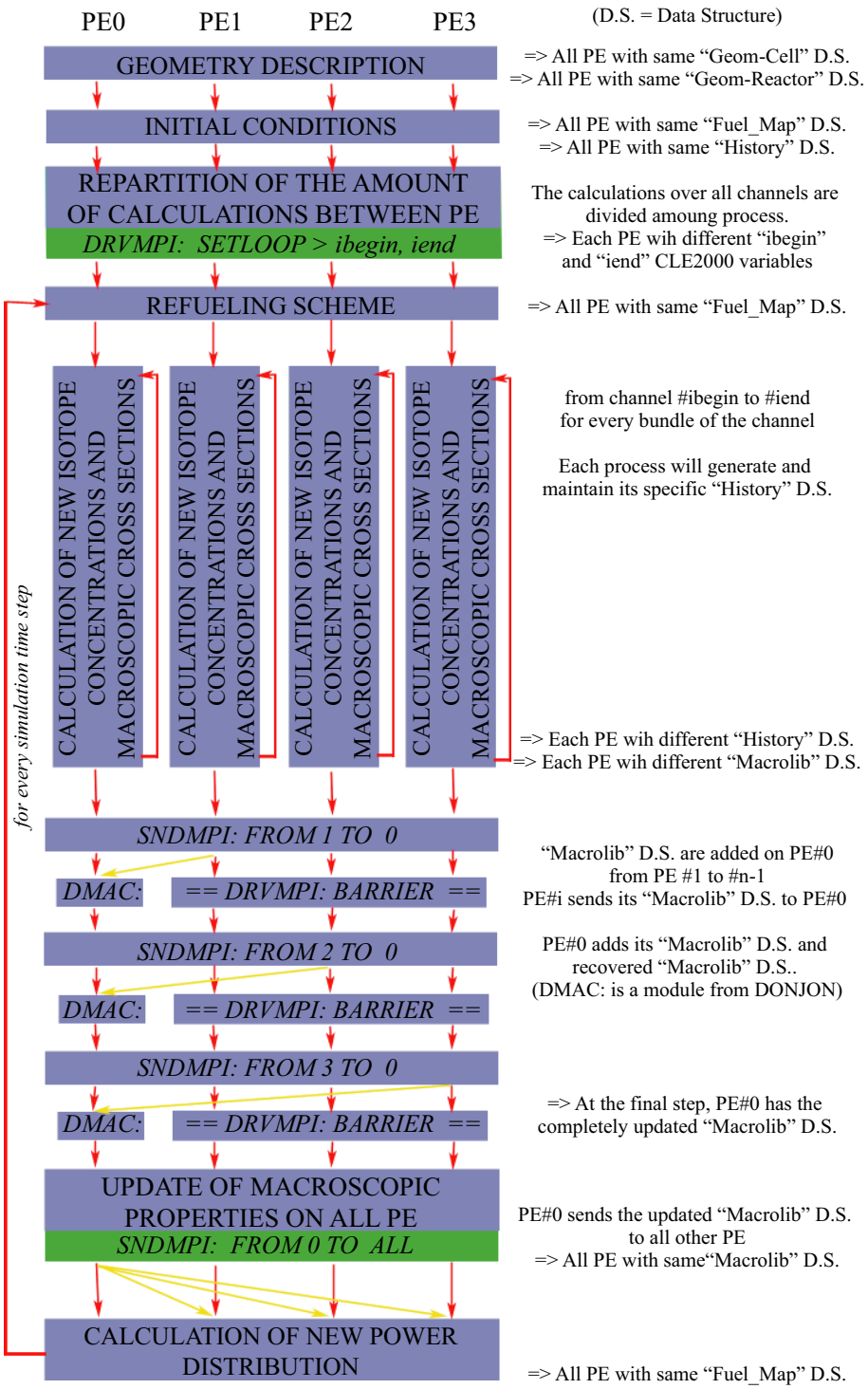

Figure 2: Parallel Algorithm for History-Based D&D Calculations

sections are integrated over the refueling period specified by an average discharge burnup and a refueling scheme. Here, we chooe the average discharge burnup in such a way that the core reactivity would be near critical. This is achieved using a slightly enriched fuel (0.9wt%) with a discharge burnup of 13000 MWd/t and a one-bundle shift fueling strategy.

The time average model generates the fuel burnups at the beginning (ω_{jk}^{BOC}) and end (ω_{jk}^{EOC}) of cycle for each bundle k in channel j. The age model assumes that burnup varies linearly with time during a cycle in every channel. Channel age is defined simply as the fraction f_j of the refueling interval elapsed since the last refueling in that specific channel. For the 16-channel cell, we have chosen a refueling order that disperses the age l_j of the fuel in a 2-D matrix:

	1	2	3	4
A	16	3	9	11
B	7	10	5	1
C	14	8	6	15
D	4	12	2	13

(4)

The resulting bundle burnups are computed as:

$$\omega_{jk}(t) = \omega_{jk}^{BOC} + \frac{(l_j - 0.5)}{16}(\omega_{jk}^{EOC} - \omega_{jk}^{BOC}) \tag{5}$$

In the channel with $l_j = 1$, bundles are at the beginning of cycle while for $l_j = 16$, bundles are at the end of cycle. Thus, a refueling sequence is defined by the channel numbering from $A1$, $l_j = 16$ to $B4$, $l_j = 1$.

Using the instantaneous model, we have evaluated the reactivity loss for depletion during one day at full power. Here, we have selected a total cell power of 40 MW, for an average channel power of 2.5 MW. These values were chosen to obtain an average bundle power of 0.6 MW, similar to what is found in a CANDU-6 reactor. Using a burnup dependent macroscopic database, the core reactivity is found to decrease by 0.7 mk per day. On the other hand the refueling of one channel increases the core reactivity by about 4.7 mk. If we assume that the reactivity effect of a refueling is independent of the sequence, a channel will need to be refueled every 6 to 7 full power days. With the channel numbering and the prescribed depletion time between two refuelings, a cell follow-up sequence was defined. Assuming that the sequence begins at 100 FPD (full power days) the full simulation period lasts 100 FPD, with 17 time steps. The first time step is just a depletion step to decrease the core reactivity around 1. This sequence of calculations will be realized using a burnup dependent macroscopic database, and the D&D history-based technique described above.

Prior to the history-based calculations, a coherent HISTORY data structure has to be generated. Using the initial bundle burnups and powers obtained using the age model and stored in the FUEL_MAP structure, a depletion calculation for each bundle is performed and the last isotopic concentrations are stored in the HISTORY structure.

The local parameters taken into account are the depletion power density provided by the FUEL_MAP structure and the fuel temperature. The latter is computed using a CANDU specific correlation based on the fuel power as:[17]

$$T_f = T_c + 0.476 * P^H + 2.267E - 4 * (P^H)^2 \tag{6}$$

where T_c is the coolant temperature, fixed to 560.66 K. The other local parameters are fixed to their nominal values.

Figure 3 shows the reactor multiplication factor during the refueling sequence. Results with the burnup-dependent cross section database form a curve around a mean value of 1.00057, while the coupled results are around a mean of 0.99854. However the two curves have the same shape, and the reactivity difference decreases slowly from 2 mk to 1.6 mk.

When comparing the maximum channel and bundle powers, one observes that the coupled version values are systematically lower that the burnup-dependent database results. Maximum channel powers decreases by a maximum of 1%, while the bundle powers drop by 1% to 3%. These results are consistent with previous studies with the feedback model on DUPIC fuels.[18]

Parallel calculations have been realized on a Linux cluster of up to 6 PIII-866 processors with 512 Mb RAM, in a distributed memory storage. The MPI version LAM 6.5.6 is used, on a 100Mb ethernet net. Transport cell calculations are distributed on 4 processors, 4 channel on each. Prior to each reactor calculation, 4 MACROLIB

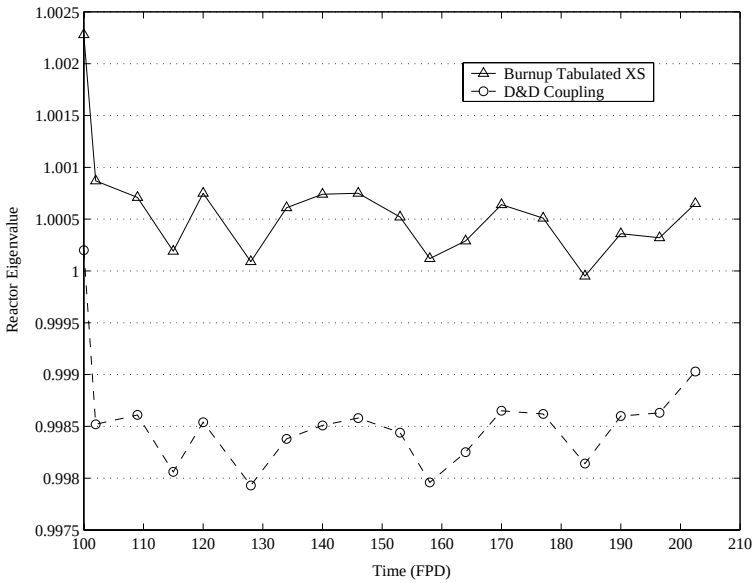

Figure 3: Follow-up core reactivity in D&D

structures are exchanged, each of 34kb size. Sequential calculations on one such processor takes 1602 minutes, around 27 hours while the 4-CPU calculations lasts only 402 minutes. Maximum speed-up is 3.985, almost ideal as expected.

Conclusion

The transport code DRAGON and the diffusion code DONJON have been coupled to develop an history based approach in the context of CANDU reactor follow-up. This method relies on a transport calculation for each fuel cell in the reactor to take care of the local parameter and refueling information.

A new module `HST:` and an HISTORY data structure have been developed to store the fuel cell isotopic concentrations, depletion information and local parameters that are defined by the FUEL_MAP structure in DONJON. The intrinsically independent transport calculations are parallelized using two dedicated modules `DRVMPI:` and `SNDMPI:`, based on MPI routines and the GAN data structures.

A simple 16-channel supercell has been defined and a refueling sequence has been followed to evaluate the new coupled code D&D. Results for both coupled and burnup-dependent properties show that the reactor eigenvalues exhibit the same variation along the simulation period. A systematic difference of 2 mk is observed. The bundle and channel powers drop when the coupled version is used, as expected. The parallel version of the coupling provides an ideal speed-up, as the transport calculations are distributed along processors.

The coupled code D&D can now be tested on more realistic case, as a CANDU-6 full core follow-up. In such a case, the number of cell calculations per refueling is nultiplied by a factor of 70. If the problem is totally scalable, as it should be, one can predict 20 full days of computations on 4 processors to realize a 16-refueling sequence. This time scale is prohibitive for current on-site calculations, but the results would provide a much needed assessment of the errors generated by various approximation methods. Moreover, the options selected in DRAGON for the cell geometry and tracking as well as the solution precision may be revisited to provide an acceptable balance between the precision of the cell calculations and the calculation time.

Acknowledgments

This work was supported in part by a grant from the Natural Sciences and Engineering Research Council of Canada and by the Hydro-Québec Chair in Nuclear Engineering.

References

[1] A. Kavenoky, "Supercomputing and Nuclear Reactor Calculations: A New Approach", *ANS Topical Meeting on Mathematics and Computation, Reactor Physics and Environmental Analysis in Nuclear Applications*, Madrid (1999).

[2] M. T. Sissaoui, G. Marleau, and D. Rozon, "CANDU Reactor Simulations Using the Feedback Model with Actinide Burnup History", *Nuclear Technology*, **125**, 197-212 (1999).

[3] G. Hotte, M. Beaudet, D. Brissette "Flux Mapping Applications Using In-core Detector Measurements for Fuel Management at the Gentilly-2 NGS", *IAEA Meeting of the Int. Working Group on Adv. Technologies for Water Cooled Reactors*, Toronto (1993).

[4] E. Varin, R. Roy, R. Baril, G. Hotte, "Validation of DRAGON/DONJON Codes Against CANDU-6 Reactor Operation Data", *ANS Topical M&C Meeting: Joint International Conference on Mathematics Methods and Supercomputing for Nuclear Applications*, Saratoga Springs, New York (1997).

[5] E. Varin, A. Hébert, R. Roy and J. Koclas, *A User's Guide for DONJON*, Institut de Génie Nucléaire, École Polytechnique de Montréal, Report IGE-208 (2000).

[6] B. Rouben and D. Jenkins, "A Review of the History-based Local Parameters Methodology for Simulating CANDU Reactor Cores", *INC'93*, Toronto (1993).

[7] B. Rouben, *An overview of Current RFSP-Code Capabilities for CANDU Core Analysis*, Atomic Energy of Canada Limited, Report AECL-11407 (1996).

[8] B. Rouben, *Description of the Lattice Code POWDERPUFS-V*, Atomic Energy of Canada Limited, Report AECL-11357 (1995).

[9] J.V. Donnelly, *WIMS-CRNL: A User's Manual for the Chalk River Version of WIMS*, Atomic Energy of Canada Limited, Report AECL-8955 (1986).

[10] B. Arsenault, J.V. Donnelly and D. Jenkins, "History-Based Calculations Using WIMS-AECL in RFSP", *20th CNS Simulation Symposium*, Niagara-on-the-Lake (1997).

[11] P.L Laughton and D.A. Jenkins, "Distributed Fuel-Management Computation Using RFSP, WIMS-AECL and PVM", *Fifth International Conference on Simulation Methods in Nuclear Engineering*, Montreal (1996).

[12] G. Marleau, A. Hébert and R. Roy, *A User's Guide for DRAGON*, Institut de Génie Nucléaire, École Polytechnique de Montréal, Report IGE-174 Rev. 5 (2000).

[13] R. Roy and A. Hébert, *The GAN Generalized Driver*, Institut de Génie Nucléaire, École Polytechnique de Montréal, Report IGE-158 (2000).

[14] B. Dionne, J. Koclas and A. Teyssedou, "Parallel Coupling of Thermal-Hydraulics and Reactor Physics for Applications in a CANDU-6", *22nd CNS Annual Conference*, Toronto (2001).

[15] E. Varin, R. Roy and T. NKaoua, "A Parallel Algorithm for the Simultaneous Solution of Direct and Adjoint Multigroup Diffusion Problems", *Lec. Notes in Computer Science, Proc. Euro PVM/MPI 98*, Liverpool (1998)

[16] W. Kropp, E. Lusk, and A. Skjellum, *Using MPI: Portable Parrallel Programming with Message Passing Interface*, MIT Press (1994)

[17] D. Rozon, *Introduction to Nuclear Reactor Kinetics*, Ecole Polytechnique (2000).

[18] D. Rozon and W. Shen, "A Parametrized Study of the DUPIC Fuel Cycle to Reflect Pressurized Water Reactor Fuel Management Strategy", *Nucl. Sci. Eng*, **138**, 1-25 (2001).