Computer Science • 16 (4) 2015 http://dx.doi.org/10.7494/csci.2015.16.4.329

Mikolaj Oettingen
Jerzy Cetnar
Tomasz Mirowski

THE MCB CODE FOR NUMERICAL MODELING OF FOURTH GENERATION NUCLEAR REACTORS

Abstract *R&D in the nuclear reactor physics demands state-of-the-art numerical tools that are able to characterize investigated nuclear systems with high accuracy. In this paper, we present the Monte Carlo Continuous Energy Burnup Code (MCB) developed at AGH University's Department of Nuclear Energy. The code is a versatile numerical tool dedicated to simulations of radiation transport and radiation-induced changes in matter in advanced nuclear systems like Fourth Generation nuclear reactors.We present the general characteristics of the code and its application for modeling of Very-High-Temperature Reactors and Lead-Cooled Fast Rectors. Currently, the code is being implemented on the supercomputers of the Academic Computer Center (CYFRONET) of AGH University and will soon be available to the international scientific community.*

Keywords Monte Carlo, Nuclear Reactors, Radiation Transport, MCB, VHTR, LFR

Citation Computer Science 16 (4) 2015: 329–350

1. Introduction

In 2002, the Generation IV International Forum (GIF) selected six innovative nuclear reactor technologies considered as the most promising concepts for the future deployment of nuclear energy sources: the Very-High-Temperature Reactor (VHTR), Molten Salt Reactor (MSR), Sodium-Cooled Fast Reactor (SFR), Supercritical-Water-Cooled Reactor (SCWR), Gas-Cooled Fast Reactor (GFR), and Lead-Cooled Fast Reactor(LFR) [10]. The proposed technologies were never before implemented for commercial applications – only laboratory-scale and half-industrial demonstrators (sometimes for defence purposes) were built to evaluate the feasibility of the systems. From a technical point of view, the decision of GIF was driven by the ability of the particular technology to improve reactor safety and reduce the production of radioactive nuclear waste. Since achieving the highest level of safety is of the utmost importance (primarily to maintain the public confidence in nuclear), the behavior of the investigated system during normal operational conditions as well as in accidental situations must be known. Therefore, the conceptual design of the Fourth Generation nuclear reactors consists of a series of mutually dependent numerical simulations of a multidisciplinary nature at different times and scales. However, among all technological tasks, the design of the nuclear reactor core is one of the most sophisticated assignments and demands usage of state-of-the-art numerical tools like The Monte Carlo Continuous Energy Burnup Code – MCB [7]. In this study, we present the main features of the MCB system developed at AGH University as well as its areas of usage in numerical simulations of the LFR and VHTR.

2. Problem statement

The nuclear reactor core consists of an array of fuel, moderator/coolant, structures, and control elements. The atomic densities of various isotopes that constitute a reactor core continuously change because of their interaction with particles (especially neutrons) and radioactive decays. The most important phenomena occur in the nuclear fuel, in which isotopes undergo fission or radiative capture reactions forming light fission products and heavy transuranic elements in the process, respectively. The changes in isotopic fuel composition during reactor operation affect the distribution of neutron flux and fission power as well as the effective neutron multiplication factor K_{eff}. The last parameter is extremely important for safety analysis because it determines neutron balance in the reactor core.

The aforementioned change in the isotopic fuel composition caused by a series of neutron interactions and decays is called fuel burnup. The detailed burnup analysis should predict fuel composition as a function of time and space, which allows us to determine the time-dependent burnup divergence in the nuclear reactor core and to assess integral safety parameters like K_{eff} or reactivity[1]. However, being able to

[1] $\rho = \frac{K_{\mathrm{eff}}-1}{K_{\mathrm{eff}}}$

solve changes in nuclear fuel composition under irradiation is a non-trivial, multipart task due to the mutual dependence of neutron flux from material composition and vice versa. To solve this problem, one needs to know the reaction rates obtained in neutron transport calculations serving later for calculating fuel isotopic composition. The isotopic composition of the irradiated nuclear fuel contains thousands of isotopes with different nuclear properties (like nuclear cross sections and half-lives) influencing the neutron flux. The newly formed isotopes interact with the neutron field, forming a non-linear chain of radioactive decays and transmutations[2] (see Fig. 1).

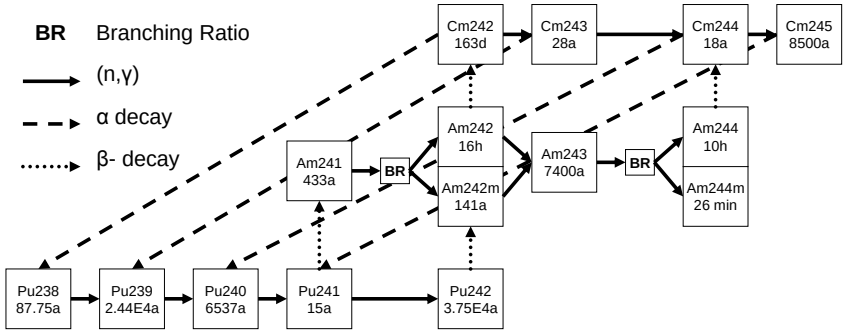

Figure 1. Example of transmutation and decay chain for plutonium, americium, and curium isotopes.

From a mathematical point of view, the first phenomenon is described by the neutron transport equation (1) derived from the Boltzmann transport equation, while the second is derived from the Bateman equation (2) [14] (see Section 4.2). Precisely, equation (1) presents the K_{eff} eigenvalue problem formulation of the neutron transport equation. A solution of the defined problem might be obtained only by performing sequential radiation transport and burnup calculations for the predefined time steps, deploying numerical tool containing modules for radiation transport and burnup calculations (like MCB).

$$(L + C)\Psi = S\Psi + \frac{1}{K_{\text{eff}}} M\Psi \tag{1}$$

where:

Ψ – the position, energy and angle dependent neutron flux,
K_{eff} – eigenvalue, effective neutron multiplication factor,
L – leakage operator,
C – scattering and absorbtion losses operator,
S – scatter-in operator,
M – fission multiplication operator.

[2]In MCB, the formation of a new isotope due to radiative neutron capture.

3. Fourth Generation nuclear reactors

Nuclear power systems characterized by the innovative features and foreseen for construction beyond 2030 are called Fourth Generation nuclear reactors [10]. As mentioned before, the GIF forum has chosen six promising systems able to fulfill the established goals of sustainability, economics, safety and reliability, proliferation resistance, and physical protection:

- Sustainability – the focal point of nuclear sustainability is to minimize the environmental impact of nuclear power generation through satisfying the strict objectives of minimizing greenhouse gas emission and the effective utilization of nuclear fuel with a minimum production of radioactive nuclear waste using the close-fuel-cycle strategy,
- Economics – the Fourth Generation nuclear systems should show a financial risk level similar to other non-nuclear technologies (even if not used for electricity generation; e.g., the production of process heat, hydrogen, or potable water),
- Safety and reliability – the Fourth Generation reactors should exhibit enhanced safety characteristics to prevent severe accidents and mitigate their consequences (incl. eliminating the need for off-site emergency response) through the use of innovative coolants, passive safety features, and enhanced training of plant personnel, among others,
- Proliferation resistance and physical protection – each plant deploying a Fourth Generation system should have high protection capabilities against terrorism and natural disasters, which (along with the goals mentioned above) is important to increase public confidence and trust in nuclear power.

3.1. Lead-Cooled Fast Reactor

The properties of liquid lead or lead-bismuth eutectic deployed in the proper manner using specific engineering features could guarantee the fulfillment of GIF goals by LFR rectors [13]. From a physical point of view, the main advantage of lead is its low neutron moderation and neutron absorption characteristic, which assures the fast neutron spectrum necessary to incinerate minor actinides and reduce radioactive nuclear waste. Additionally, the design assumes decay heat removal only by natural convection in accidental situations, which increases safety. Moreover, lead strongly reflects neutrons, shields against γ-ray radiation, and does not interact exothermically with water or air. The last property provides an opportunity for the elimination of the intermediate cooling loop, leading to plant simplification, a reduction of capital costs, and possibly increasing reliability. The neutronic characteristics of the LFR core also allow for long-life cores, resulting in a significant extension of reactor cycle, which minimizes the proliferation risk and diversion of weapon-usable heavy isotopes. Lead also retains volatile contaminants produced in the fission process and during activation of the coolant, especially isotopes of cesium and iodine [8].

3.2. Very-High-Temperature Gas-Cooled Reactor

The HTR reactors in the Fourth Generation nuclear system are designed to be capable of achieving high coolant outlet temperatures – according to the design envelope, at least 850 °C. The increase in temperature compared to standard Light Water Reactors allows for more-efficient power generation using the Brayton cycle and better thermal conditions for processing heat generation for industrial processes like synfuel or hydrogen production. Low-temperature process heat at about 600 °C could be applied for pulp or iron manufacturing. The crucial design parameters rating HTR as an innovative nuclear system are: helium coolant, graphite moderator, and refractory coated particle fuel – TRISO[3]. The combination of the three aforementioned features provides inherent safety characteristics in case of accidental situations such as LOCA (Loss of Coolant Accidents), with no risks for large nor early releases. The high thermal capacity of graphite assures better decay heat removal and core stability in high temperatures. The TRISO fuel particles assure fuel stability at high temperatures with the ability to retain radioactive fission products. Helium was chosen as a coolant because of its chemical and radiological inertness [17].

3.3. MCB modeling of LFR and HTR reactors

The advanced methodologies for the design of the VHTR and LFR reactor cores were developed using the MCB system. The simulation of VHTR was performed in the frame of PUMA (Plutonium and Minor Actinides Management in Thermal High Temperature Reactors), while LFR simulation was performed in the frame of ELSY (European Lead-cooled System projects). Both projects were sponsored by the European Commission within 6^{th} EURATOM framework program. Lastly, the MCB code was used for the modeling of LFR in the frame of LEADER - Lead-cooled European Advanced Demonstration Reactor project within 7^{th} EURATOM framework program. Work on the VHTR reactor is continued in the HTRPL project sponsored by the National Center of Research and Development, Republic of Poland. The methodologies consist of many individual steps upon which system characteristics are estimated, which later serve the safety analysis. The main steps of the methodologies contain:

- transformation of the engineering geometry to the high-resolution computational geometry of the MCB model,
- linkage of the nuclear data libraries and input of the isotopic material compositions,
- definition of burnup zones and times steps for Transmutation Trajectory Analysis (TTA),
- implementation of the numerical constraints: cut-off levels of numerical procedures, numerical precision, definition of neutron source, etc.,

[3]Tristructural Isotropic coated fuel particle.

- initiation of scoring functions for events of interest; e.g., neutron absorption or fission,
- multi-scale Monte Carlo simulations,
- visualization of the results and rerun for the new set of input parameters.

The analysis performed initially in the ELSY and continued in the LEADER projects included burnup calculations in a few-batch fuel reloading schema, assessment of the equilibrium fuel composition, assesement of power distribution with its evolution with burnup, and modeling of the control rod operation. Modeling of the HTR reactor in the frame of PUMA and HTRPL projects focuses on the design of the high-resolution, 3D, full-core computational geometry for neutronic and thermo-hydraulic calculations.

3.4. Computational Challenges

In this section, we summarize the computational challenges in the development of the MCB system towards numerical modeling of the Fourth Generation nuclear reactor. In the conceptual design phase, knowledge about system behavior might only be obtained using a numerical simulation. At this stage, empirical knowledge is simply unavailable. Therefore, designers applying numerical tools should always strive for the highest possible precision of calculations, which should serve the experimental validation in the future and (later) improve accuracy. The numerical models of the Fourth Generation nuclear reactors are characterized by their large spatial heterogeneity and complicated material composition, which necessitates the use of HPC computers to reconstruct the behavior of the system with the desired precision. The MCB system is currently being implemented on the supercomputers of the ACK CYFRONET AGH [9]. This will help solve the following computational problems:

- Storage: every MCB run produces a few output files demanding disk storage resources – for one, mid-complicated professional run in the order of magnitude of GB. In addition, the MCB uses a few sets of nuclear data libraries usually available to all users – one professional ASCII library with nuclear data demands tens of GB. The ACK CYFRONET provides enough storage space for all necessary data,
- Execution time: The Monte Carlo MCB methods use a quiet simple theoretical background to solve complicated numerical problems, but their drawback is the extended execution time. A decrease in execution time might be achieved using multi-core calculations by means of Message Passing Interface (MPI) at the supercomputers of ACK CYFRONET,
- Practical implementation: For new users, the MCB installation and execution process seems to be extremely complicated. The former could be simply solved by providing compiled MCB executables, pre-defined launching scripts, and modules for all MCB users in the assigned workspace. The latter could be significantly simplified by uploading a users manual (in a pleasant form) to the web portal of the PLGRID project [19].

4. MCB

MCB is a general-purpose code dedicated to simulations of radiation transport and radiation-induced changes in matter. The accurate numerical simulation in both areas provides an exact description of the physical effects appearing in the nuclear reactor core. The MCB entirely integrates the commercial Monte Carlo transport code A General Monte Carlo N-Particle Transport Code (MCNP) [28] and the novel Transmutation Trajectory Analysis Code (TTA) at the level of the FORTRAN source code. MCNP subroutines are used for radiation transport simulation, while the TTA automatically forms and analyzes transmutation and decay chains for the nuclide density evolution in time function. The applied MCNP-TTA coupling schema is presented in Figure 2. The code is a versatile numerical tool dedicated to simulations of advanced nuclear systems, including the Fourth Generation nuclear reactors. The main added value compared to other numerical tools is its multipurpose character allowing numerical modeling of many physical effects in one simulation run. The necessity for external coupling with other numerical tools for analyzing the nuclear reactor core was significantly reduced. The main competition to the TTA method implemented in MCB are the assorted variations of the exponential matrix method; e.g., the Chebyshev Rational Approximation Method (CRAM) implemented in SERPENT code [25].

All MCNP features are available in the MCB, and some new ones were added. In the MCB calculations, one can obtain the following parameters: neutron multiplication factor and neutron multiplication, standard deviations of neutron multiplication factor, neutron source importance of inducing fission and non-fission reactions, neutron source strength, energy deposition per source neutron/proton, potential dose, material activity, neutron flux distribution, heating distribution, decay heat[4] distribution, afterheat[5] distribution, reaction rates distribution, transmutation constants, transmutation trajectories and transition probabilities, fission product distribution over burnable zones, activations product masses, actinide masses, density of fissioning, proton current (in case of modeling sub-critical Accelerator Driven System-ADS), and many others.

The MCB code was widely used as a primry numerical tool for reactor physics calculations in six European Commission projects within EURATOM framework programs:

- PDS-XADS – Preliminary Design Studies of an Experimental Accelerator-driven System (2001–2004),
- EUROTRANS – European Research Program for the Transmutation of High Level Nuclear Waste in an Accelerator Driven System (2005–2009),
- PUMA – Plutonium and Minor Actinides Management in Thermal High Temperature Reactors (2006 – 2009),

[4]The heat released as a result of radioactive decay.

[5]The heat released as a result of radioactive decay by the radioactive fission products and transuranic elements remaining in nuclear reactor after it has been shut down.

- ELSY – European Lead-cooled System (2006–2009),
- LEADER – Lead-cooled European Advanced Demonstration Reactor (2010–2013),
- FREYA – Fast Reactor Experiments for Hybrid Applications (2011 – ongoing).

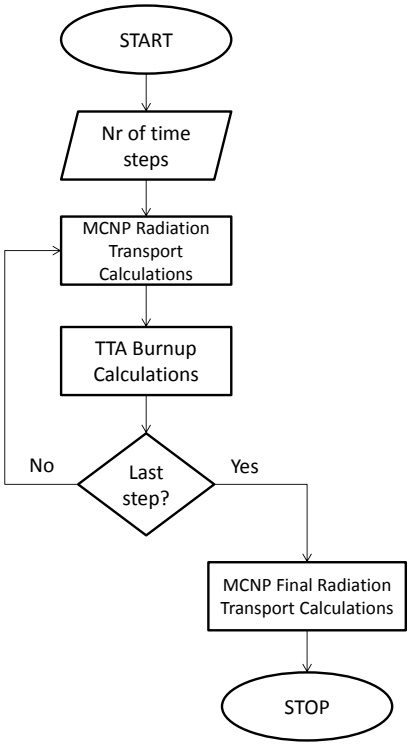

Figure 2. MCNP-TTA coupling algorithm.

The development of MCB aims at creating a multi-physics code that could provide an exact description of the nuclear reactor or other nuclear systems during normal operational conditions and accidental situations. The multi-physics modeling refers to the integrated estimation of many parameters coming from the different scientific areas; namely: radiation transport, burnup, thermo-hydraulics, and structural analysis. Such an integral approach is more computer-time consuming; therefore, the latest version of the MCB was coupled with the Message Passing Interface (MPI) multiprocessing interface. The MCB is still under development at AGH University of Science and Technology, Krakow, Poland, Department of Nuclear Energy.

4.1. Program flow chart

The calculation of nuclide density time evolution is performed using the coupling algorithm shown in Figure 2. First of all, at the beginning of an arbitrary time step,

the MCNP subroutines of the MCB calculate contribution to the neutron spectra from each sampled particle in the selected geometrical regions of the numerical model (called either burnup or fuel zones). The burnup zones are filled with materials representing nuclear fuel that undergoes nuclear reactions or radioactive decay. On this basis, decay and reaction probabilities for each possible channel are assigned, and reaction rates are calculated. In the second processing step, the code automatically forms material- and time-dependent transmutation and decay chains. Afterwards, the trajectory analysis process is used to expand chains into a series of transmutation trajectories, and the formed Bateman equations are solved using the linear chain method [6], which results in new nuclide densities at the end of a time step. Next, neutron spectra for new nuclide densities are calculated and new decay and reaction probabilities are assigned. The whole process is repeated for all given time steps. The calculation procedure for the first time step is different. The so-called exploratory step is performed for the assessment of the emerging nuclides that could appear due to the interaction with a particle or decay.

4.2. Background of the linear chain method

The Bateman equations describe the time evolution of nuclide density. The linear or serial decay chains are expressed by a set of first-order differential equations. However, the complexity of the nuclear transmutation and decay system leads to the formation of a non-linear chain structure. The MCB code decomposes the complex non-linear chain structure into a set of individual linear chains (as presented in Figure 3). After simplifying the non-linear chain, the code assigns and solves the individual Bateman equation for each corresponding linear chain. The solution of the whole non-linear chain structure corresponds to the superposition of all linear chain solutions. This is called the linear chain method, or TTA.

Equation (2) corresponds to the simple linear Bateman equations of radioactive decay for n number of nuclides, where λ_i is the decay constant for the i-th nuclide while b_i is the branching of i-th nuclide decaying into *[i-1]*-th nuclide.

$$\frac{dN_i}{dt} = b_i \lambda_{i-1} N_{i-1} - \lambda_i N_i \tag{2}$$

If the initial boundary conditions assume a zero concentration of all daughter isotopes at time zero i.g. $N_1(0) \neq 0$ and $N_i(0) = 0$ for i>1 , the concentration of the n-th nuclide after time t is expressed as:

$$N_n(t) = \frac{N_1(0)}{\lambda_n} \sum_{i=1}^{n} \lambda_i \alpha_i exp[-\lambda_i t] \tag{3}$$

where:

$$\alpha_i = \prod_{j=1, j\neq i}^{n} \frac{\lambda_j}{\lambda_j - \lambda_i} \tag{4}$$

As mentioned above, the nuclear transmutation/decay system is more complex due to the branching and formation of transmutation loops. Thus, an additional more-sophisticated mathematical set-up was implemented in the MCB code [6].

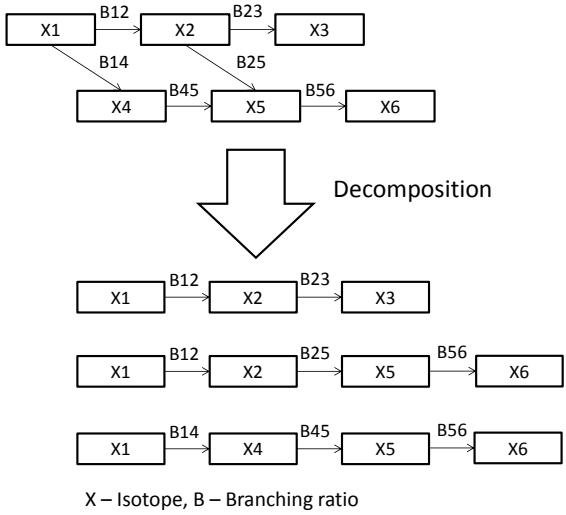

Figure 3. Linear chain method.

4.3. Definition of transmutation system

MCB could work in the pure particle transport mode or in the burnup mode. The control card BURN specified in the input file launches subroutines calculating the time evolution of nuclide density. For burnup calculation, the user is obliged to specify physical constraints over a transmutation system. Apart from the system geometry, the specifications of the materials, control parameters, and time steps must be done.

There are three kinds of material definition in the MCB code. Transport and burnup materials are characterized by the density, atom, or weight fraction of compounding isotopes and cross-section libraries. The burnup materials are used for reaction-rate calculations, and their density changes due to the neutron interactions and decays in the particle field. The transport materials are used just for the particle transport calculations. The last type of materials (residual materials) are used for calculations of the transmutation transitions. They do not contribute to the buildup of new nuclides and do not have an assigned density. The material classification was introduced in order to facilitate the input file; e.g., in the pure neutron transport problem, the definition of burnup materials is needless.

Time intervals might be defined in two ways: as a series of the individual irradiation periods, or as cumulative values. Both definitions are interchangeable and it is up to the user which one to apply. Some additional actions (especially those linked

to material processing) may be activated at the specified time points. The process of material treatment refers to the material removal, change, addition, admixing, and shuffling. Using material processing cards, one can simulate control rod movement, fuel shuffling, coolant density changes, insertion of experiments, and other actions.

The control parameters concern three main issues:

- time control; e.g., the longest-allowed duration of irradiation period,
- control of numerical procedures; e.g., cut-off limits of nuclide density and dose printout,
- control of burnup process.

The last two parameters are worth noting because they are essential for the calculation efficiency and numerical solution of the Bateman equations. Firstly, the decay half-life threshold for the fast decay treatment indicates nuclides that decay immediately after appearing. This parameter was introduced to shorten calculation time and to only take care of nuclides that are significant for the transmutation chain formation. Secondly, the discrimination level of the nuclide-to-nuclide transmutation transition is used for the truncation of the trajectory extension process [6].

To obtain the power distribution in burnup regions, the fissionable system has to be normalized. There are two standard normalization procedures. The system may be normalized to total thermal power or to the strength of an external neutron source. The first option is used in eigenvalue calculations, while the second in source calculations that do not demand estimation of the system eigenvalue. In the case of the subcritical system; e.g., Accelerator Driven System (ADS), normalization of the proton beam current is also possible. The power (heating) calculations may be based on KERMA[6] factors[7] or on the total energy released per fission – Q_{fiss} values included in the standard transport cross-section libraries [18].

4.4. Nuclear data

For the neutron transport calculations, the MCB may use various evaluated cross-section libraries processed for the arbitrary temperature step in ENDF [12] format; e.g., JEF, JEFF, JENDL, ENDF-B/VI-8, EAF[8], and others. The additional sets of nuclear data are required for the simulation of nuclide formation due to the natural decay and nuclear reactions. For this reason, MCB was equipped with four extra library files containing data from a few different sources:

- TOI.LIB file with decay scheme for about 2400 nuclides and dose data for about 738 nuclides,
- XS.LIB file with one-group cross-section libraries containing energy-integrated ratios of the nuclide formation in excited state due to neutron capture (n, 2n) reaction, and nuclear decay,

[6]KERMA – Kinetic Energy Released per Mass Unit.

[7]Factors depend on the pre-processing of the library usually by the means of NJOY code.

[8]Evaluated Nuclear Data File, Joint Evaluated File, Joint Evaluated Fission and Fusion, Japanese Evaluated Nuclear Data Library, European Fusion File.

- ISB.LIB file with branching ratios to the ground and metastable state as the function of the incident neutron energy for Am242m and Am244m, based on the Mann and Schenter model [16],
- FY.LIB file with incident energy depended fission product yields for 36 nuclides based on Wahl evaluation [27].

5. Implementation

The following section shows the current status of the MCB system as well as areas of possible improvements achieved by the implementation at the HPC computers. However, we will first present some important information about the practical usage of the code and its structure. This should help one understand the general logics of MCB.

5.1. Structure

The coupling between MCNP and MCB subroutines is realized in accordance to the MCNP developers guide [29]. According to the new procedure of code development, the code developer creates a patch for MCB and applies it for building MCB extensions on the existing MCNP source directory – see Figure 4.

The radiation transport source code is in Source/src/, where MCB source files are also placed. The subroutines and data modules that constitute the code can be divided into three classes with regard to the status of their modifications: unchanged original MCNP files, modified MCNP files, and MCB specific files. The modified files are printed in a boldface font. Appropriate source file names are listed in the following files: FILE.list, FILE_MCB.list, FILE_MCB+MODIFIED.list while dependency on existing MCNP modules in Depends file, which is necessary for the compilation using the MAKE utility [11]. The individual MCB burnup subroutines are placed in the same folder and are depicted as *.F90 in Figure 4. The MCB code can run on standard UNIX platforms as well as on LINUX-PCs. In both cases, the PVM(Parallel Virtual Machine) or MPI options of parallel execution are available. The source code can be compiled by one of the FORTRAN90 compilers (so far, many versions of INTEL and PORTLAND compilers were successfully tested) that apply the dynamic allocation of RAM. The MCB is compatible with the MCNP code and can work in the pure MCNP mode run while keeping newly introduced options of material definitions. MCB can be interrupted and restarted using the backup file; the same is true in the case of system malfunction during a run. The executable name of MCB is configured in Source/config/Code_info.gcf. The name depends on the chosen installation option as follows:

- sequential version – mcb5,
- parallel mpi version – mcb5.mpi,
- parallel pvm version – mcb5.pvm.

```
Source/
      Makefile
      install
      config/
            Makefile
            Code_info.gcf
            …
      CVF/
      datasrc/
      dotcomm/
            src/
                  Makefile
                  internals/
                  *.F90
                  mpi/
                  pvm/
      src/
            Makefile
            FILE.list
            FILE_MCB.list
            FILE_MCB+MODIFIED.list
            Depends
            *.F90
            …
      X11R6/
Testing/
      Regression/
            Makefile
            Inputs/
            Templates/
      config/
            test_options.mk
```

Figure 4. MCNP source directory after integration with MCB routines.

Figure 5 presents the basic flow diagram of MCB reflecting its full integration with the MCNP code. The subroutines in the normal font show modified MCNP routines, while routines in bold show the newly added burnup routines. In the detail description of MCNP routines, one can find in the developers guide [29]. In this study, we briefly describe the functions of the most important MCB subroutines:

FASTD: Fast decay analysis module, which makes an analysis of nuclear data concerning radioactive decay and checks its consistency and data availability in the library file. The necessary pointers from the nuclide list to the cross-section index table and to other tables are formed as well as the table offsets for dynamically allocated storage are finally completed to reserve possible storage for nuclide densities, reaction rates, etc.

STEPRUN: this routine controls the program flow and time-step procedure depending on the obtained results that are cross-checked with the user requirements. The time-step procedure is done with the subsequent execution of PREPARE, MCRUN, and PROCEED routines. The routine consists mainly of calls to data-processing routines or calculation routines.

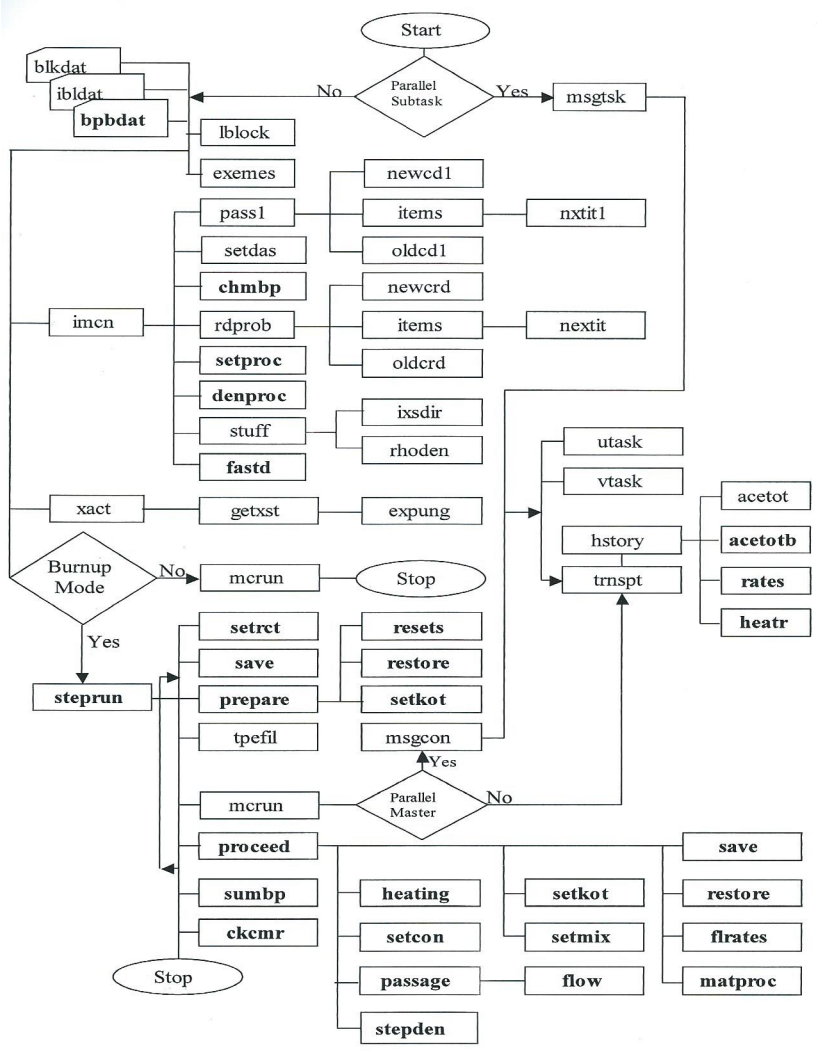

Figure 5. Flow diagram of the MCB system [5].

SETRCT: this routine is responsible for reaction cross-section processing and preparation for the initial MCNP run. The reaction list is established on the basis of specifications present in the cross-section files. At the end, the data is saved on a restart file that is later used for a problem restart in the following time steps.

HEATING: system heating calculations, which are calculated by taking into account the contributions from nuclear energy release. Later, the necessary reaction rates are converted into reaction probabilities in the SETCON routine, where the parameters of the Bateman equations are calculated as well. The contributions to

reaction rates are scored by the RATES routine, while those to heating are scored in the HEATR routine.

MATPROC: material processing subroutine for material admixing, removal, shuffling, and other actions. Material processing is grouped into batches using subroutines SETPROC and DENSPROC.

PASSAGE: this routine calculates nuclide-to-nuclide transmutation transitions and uses them to calculate nuclide density evolution in subroutine STEPDEN.

SUMBP: this routine calculates and prints out a summary of system performance to the output files. In the cases when calculation results or system setup need attention, the program prints out a warning message. For those purposes, both the output files and console are used.

5.2. Storage

The results of MCB modeling are printed out to a series of output files that contain the results of every step of calculation. Two types of output files exist: standard MCNP files related to the particle transport calculations, and MCB files related to the burnup calculations (as shown in Figure 6). In addition, the MCB system reads two types of libraries containing cross-section data (XSLIB) and radiological nuclide properties (BPLIB). Standard MCNP output files:

- outp: output file containing summary results from neutron transport calculations,
- runtpe: the binary file containing information needed to restart a problem in case of system malfunction,
- srctp: file containing the final spatial distribution of fission neutrons from the last run, which may be used in subsequent runs to decrease computation time.

MCB output files:

- bout: the main output file with the burnup calculations that contains the most important summary results, both from transport and burnup modules,
- btab: file with calculation results in the MATLAB format. It allows the user to swiftly post-process the data or make plots using the MATLAB environment,
- bmes: detailed information files with data related to burnup calculations. The file is very large, but it can be reviewed easily by searching for break lines that contain the following string: ####.

The code creates additional output files depending on input parameters specified by users. The files listed above are the default files almost always produced in the simulation.

The professional design of the particular nuclear system constitutes a series of simulation runs with many varying parameters. One run using mid-complicated numerical model of LFR or VHTR produces output files that demand a few-GB disk storage. The MCB test performed using the advanced numerical model at the level of fuel assembly shows a disk-storage demand of about 3GB for one simulation run:

- outp file – 54 MB (1.79%),
- srctp file – 8.4 MB (0.28%),
- runtpe file – 753 MB (24.92%),
- bout file – 7.3 MB (0.24%),
- btab fiel – 50 MB (1.65%),
- bmes file – 2.1 GB (69.5%),
- remaining files – 50 MB (1.62%)

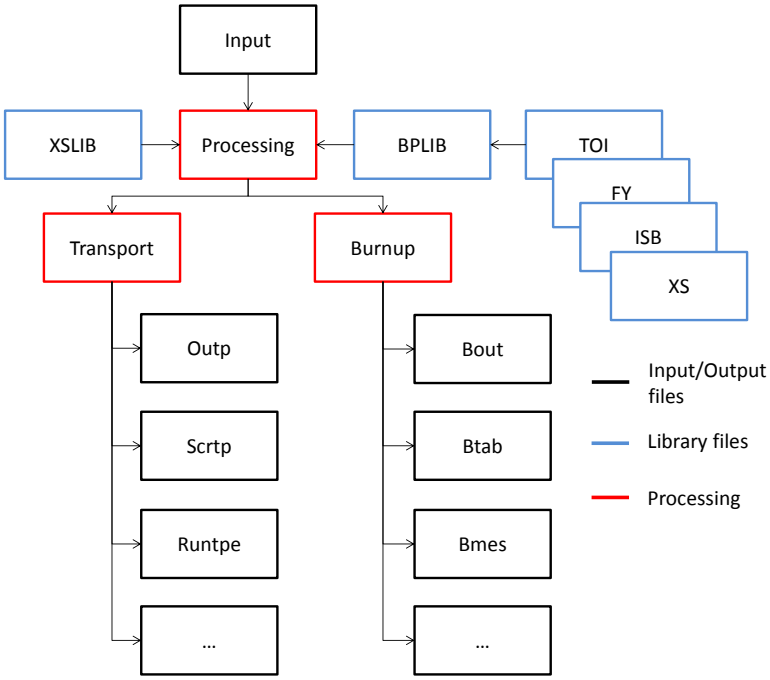

Figure 6. Output/Input file structure in MCB modeling.

The whole series of calculations demands tens, if not hundreds, of GB for interim storage, until the final elaboration takes place using post-processing software like MATLAB. In addition, the designers of the nuclear system usually use a few types of nuclear data libraries for benchmarking purposes. The standard ASCII library files demand a few tens of GB. The libraries may also be modified using dedicated software at the request of the user [15]. In this case, storage demand increases tremendously. The HPC clusters are usually equipped with data management software, which could serve the systematization of the most important simulation results and their long-term storage and backup. They also contain a large amount of disk storage space dedicated to data storage. This kind of solution is not readily available on standard LINUX or WINDOWS PC platforms due to limited storage capability. Moreover, the

dedicated utilities may be used for the creation of a nuclear data library database easily accessible by all users, without the demand of continuous and time-consuming library modifications.

Finally, it is worth mentioning the statistical analysis of the Monte Carlo process, which requires repetition of the whole simulation with just a different initial random seed. These kinds of tests are extremely demanding in disk storage and computational power. Each simulation produces a few GB, so the one fully representative test with several external replicas corresponding to the individual random seeds will produce hundreds of GB.

5.3. Parallel Monte Carlo

The term Monte Carlo refers to the simulation of the neutron transport rather than the whole calculation process. The other analyses performed after Monte Carlo transport calculations (like burnup or thermo-hydraulics calculations) might be based on a different mathematical apparatus. Nevertheless, the Monte Carlo transport calculations are the initial step in the numerical analyses of the nuclear system and play a crucial role in the whole simulation.

The Monte Carlo transport calculations consume most of the CPU time during an entire MCB run. Therefore, the MCB code could be executed in parallel mode using PVM or MPI utilities. The former is implemented due to historical reasons and is not used nowadays. The latter constitutes the new standard for particle transport calculations in MCNP/MCB systems. The Monte Carlo particle transport calculations are characterized by two crucial features, making it ideal for parallel computing. Firstly, disk storage for the 3D mesh geometry is not required because the code uses compact 3D combinatorial geometry. Secondly, the computed particle histories[9] are independent of each other. Therefore, the parallelism of the particle transport might be based on particle histories. Simply, the total number of particle histories defined by the user is divided among processors using a master/slave algorithm. The second feature indicates to the generic parallel nature of the Monte Carlo algorithm, because the slave processes does not need to exchange any information with each other. In the standard MCNP/MCB implementation, the master process is responsible for the distribution of problem data and collection of the results by means of an MPI interface, while the slave processes concurrently compute particle histories using OpenMP threads – see Figure 7. The main concerns to be addressed in the parallel execution of the MCNP/MCB system are fault tolerance and load balancing [4].

The first problem is solved by having the slaves rendezvous periodically with the master (as shown in Figure 8). At the moment of rendezvous, the calculations performed by the slaves are stopped, and the results are sent to the master. The master receives data from each slave, combines them into a combined result, and makes backup-producing dump files with a restart information. The dump files can

[9]The term history refers to the simulation of the initial particle and its "progeny"

be created after computing either N particle histories or after a specified computational time, depending on input parameters specified by the user. In case of system malfunction, the most recent dump-file (checkpoint file) can be loaded to continue the calculations starting from the moment of system failure. The efficiency of this approach depends on the rendezvous and computation time. The former, in turn, is related to the number of histories per task, while the latter to the amount of tally data, latency, and bandwidth of the message passing.

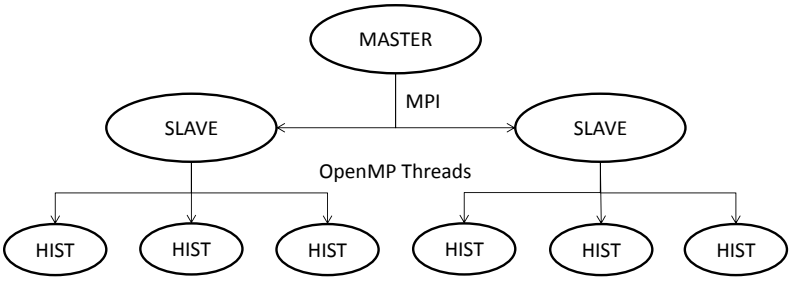

Figure 7. Monte Carlo Parallelization.

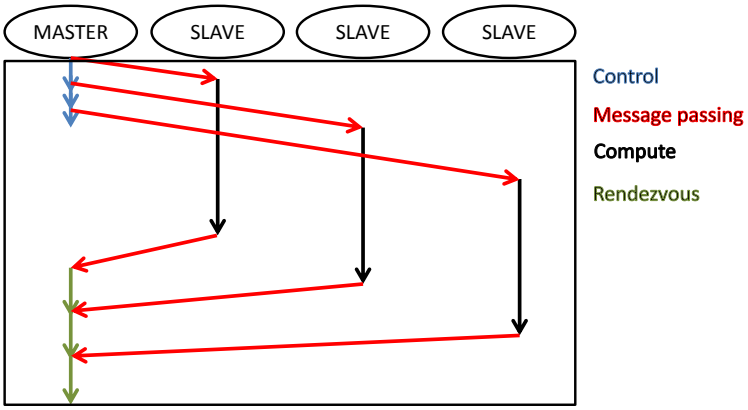

Figure 8. Fault tolerance.

During MCNP/MCB execution at workstation clusters, the problem of proper load-balancing might appear. This may be caused by the varying node performances over time due to simultaneous interactive work at the time of computation or simply by different CPU speeds. To solve this problem, the self-scheduling algorithm presented in Figure 9 was applied [3]. The total number of histories is divided into several moderate-sized chunks. The number of chunks always have to be greater than number of slave processors. If a slave is idle, it asks the master for a new chunk of work. The master sends the new chunk and waits for a new request. In this way,

the faster slave nodes process more chunks of work than slower nodes. The process is repeated until all histories are finished.

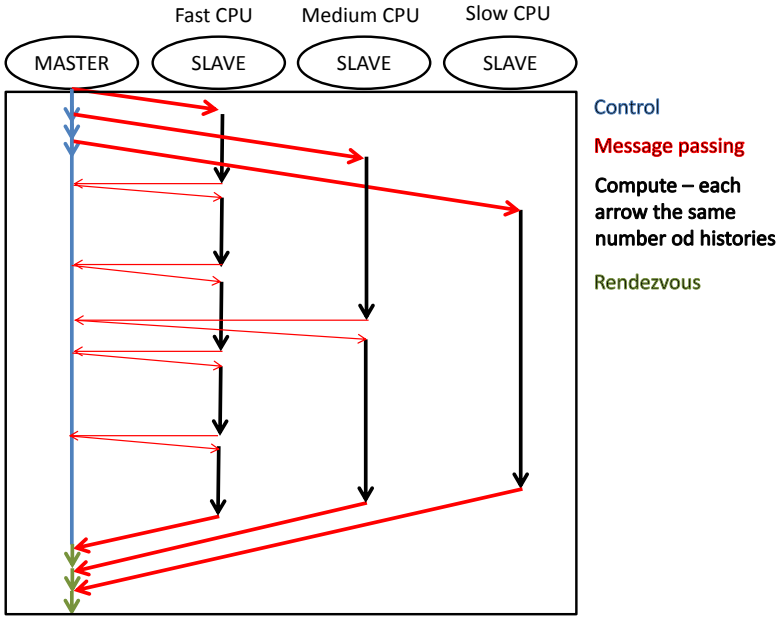

Figure 9. Load balancing.

The former section presented the current techniques implemented in the MCNP/MCB system for parallel computing in order to decrease execution time. However, the parallel algorithms for particle transport in the original MCB were never fully tested using a large number of CPUs. Till now, the code was tested in a rather ad hoc manner with a limited number of CPUs (maximally, about 40) using various processor types and workstation clusters in the frame of projects listed in section 3.3. The preliminary results have shown rather good scalability of the code. However, only the implementation on the HPC of the ACK CYFRONET gives us the opportunity to test the code using a large number of CPUs and professionally assess its scalability in the available computational infrastructure. This, in turn, will allow us to execute the numerical models of the Fourth Generation nuclear reactors characterized by large spatial resolution with reasonable execution time. The first results using the MPI interface for 12 Intel Xeon 2.26 GHz cores and numerical test-model of the fuel assembly mentioned in section 5.2 shows a walltime of about 28h with final precision of K_{eff} of 25 pcm (fully acceptable, from scientific point of view). The general assessment of the MCB performances will be provided after its final implementation on the supercomputer of ACK CYFRONET AGH.

6. Conclusions

We have presented the novel Monte Carlo code MCB dedicated to numerical simulations of radiation transport and burnup. The code provides the full set of parameters necessary for a detailed analysis of Fourth Generation nuclear reactors, including evaluation of the key safety parameters and optimization of the core performance. From a computational point of view, the implementation of MCB on the supercomputers of the ACK CYFRONET AGH will increase code performance and extend its area of use:

- the availability of a large number of CPUs will allow the execution of numerical problems with complex, full-heterogeneous and high-resolution geometry in a reasonable time using the implemented master/slave algorithm for parallel computing,
- the implemented master/slave algorithm performs rather well, but additional performance tests using a large number of CPUs are required to assess code scalability and detect possible bottlenecks,
- the large storage capability allows us to simultaneously simulate several different cases of the same numerical model with varying input parameters to create a result database for further post-processing,
- the required nuclear data libraries, as well as the tools for their processing, may be stored in the assigned work space – this decreases demand for the continuous processing of nuclear data libraries upon a particular numerical problem,
- the various environments developed and deployed in the PLGRID infrastructure, like RIMROCK [20], QosCosGrid [21], SCALARM [22], GridSpace [23], and others might be applied for facilitating MCB parametrical studies,
- additional utilities available at the cluster workstation, like Modules [26], PBS-Portable Batch System [2], Bazaar-resource management system [24], Confluence-team collaboration software [1], and others will allow efficient execution of the MCB and support buildup of the several numerical models to increase the number of code users.

To sum up: in this paper, we have shown the current status of the MCB system in modeling of the Fourth Generation nuclear reactors and the advantages associated with its implementation on the supercomputers of the Academic Computer Centre CYFRONET of the University of Science and Technology in Krakow in the frame of the PLGRID NG project.

Acknowledgements

This research has been partially supported by the European Regional Development Fund program no. POIG.02.03.00-12-138/13 as part a of the PLGRID NG project and associated PLGRID Infrastructure. Moreover, partial financial support of this study through the funds of the Polish Ministry of Science and Higher Education is kindly acknowledged.

References

[1] Atlassian: `https://www.atlassian.com/software/confluence`.

[2] Bayucan A.: *PBS Portable Batch System*. Numerical Aerospace Simulation Systems Division NASA Ames Research Center, 1998.

[3] Brown F., Goorley T., Sweezy J.: MCNP5 Parallel Processing Workshop. In: *ANS Mathematics and Computation Topical Meeting, Gatlinburg, USA, LA-UR-03-2228*, 2003.

[4] Brown F., William M.: High Performance Computing and Monte Carlo. In: *American Nuclear Society Winter Meeting, LA-UR-04-4532*, 2004.

[5] Cetnar J.: *Development and Applications of MCB – Monte Carlo Continuous Energy Burnup Code*. AGH University, Faculty of Physics and Applied Computer Science, Poland, 2006.

[6] Cetnar J.: General Solution of Bateman Equations for Nuclear Transmutations. *Annals of Nuclear Energy*, vol. 33, pp. 640–645, 2006.

[7] Cetnar J., Gudowski W., Wallenius J.: MCB: A continuous energy Monte Carlo burn-up simulation code. In: *Proceedings of Actinide and Fission Product Partitioning and Transmutation, EUR 18898 EN, OECD/NEA 523*, 1999.

[8] Cinotti L., Smith C.F., Sekimoto H., Mansani L., Reale M., Sienicki J.J.: Lead-cooled system design and challenges in the frame of Generation IV International Forum. *Journal of Nuclear Materials*, vol. 415, pp. 245–253, 2011.

[9] CYFRONET: `http://www.cyfronet.krakow.pl`.

[10] DOE: *A Technology Roadmap for Generation IV Nuclear Energy Systems*. U.S. Department of Energy, USA, 2002. GIF-002-00.

[11] Foundation Free Software: `http://www.gnu.org/software/make`.

[12] Herman M., Tarkov A.: *ENDF-6 Formats Manual Data Formats and Procedures for the Evaluated Nuclear Data File ENDF/B-VI and ENDF/B-VII*. National Nuclear Data Centre Brookhaven National Laboratory, USA, 2009. CSEWG Document ENDF-102 Report BNL National Nuclear Data Centre Brookhaven National Laboratory-90365-2009.

[13] IAEA: *Liquid Metal Coolants for Fast Reactor Cooled by Sodium, Lead, and Lead-Bismuth Eutectic*. International Atomic Energy Agency, Austria, 2012. IAEA Nuclear Energy Series No. NP-T-1.6.

[14] Isotalo A.: *Computational methods for burnup calculations with Monte Carlo neutronics*. Ph.D. thesis, Aalto University, School of Science, Finland, 2013.

[15] MacFarlane R.: *The NJOY Nuclear Data Processing System, Version 2012*. Los Alamos National Laboratory, LA-UR-12-27079 Rev, USA, 2013.

[16] Mann F., Schenter R.: Calculated Neutron Capture Cross Sections to the Americium Ground and Isomer States. *Nuclear Science and Engineering*, vol. 63, pp. 242–249, 1977.

[17] Moses D.L.: *Very High-Temperature Reactor (VHTR) Proliferation Resistance and Physical Protection (PR&PP).* Oak Ridge National Laboratory, USA, 2010. ORNL/TM-2010/163.

[18] Oettingen M.: *Validation of fuel burnup modelling with MCB Monte Carlo system using destructive assay data from Ohi-2 PWR.* Ph.D. thesis, AGH University, Faculty of Physics and Applied Computer Science, Poland, 2014.

[19] PLGRID: `https://portal.plgrid.pl`.

[20] PLGRID: `https://submit.plgrid.pl/jobs`.

[21] PLGRID: `http://www.qoscosgrid.org/trac/qcg`.

[22] PLGRID: `http://www.scalarm.com/`.

[23] PLGRID: `http://dice.cyfronet.pl/products/gridspace`.

[24] PLGRID: `https://bazaar.plgrid.pl/BazaarUP/org.cyfronet.bazaar.BazaarGwtGui/BazaarGwtGui.html`.

[25] Pusa M., Leppänen J.: Computing the Matrix Exponential in Burnup Calculations. *Nuclear Science and Engineering*, vol. 164, pp. 140–150, 2010.

[26] Sourceforge: `http://modules.sourceforge.net`.

[27] Wahl A.: Nuclear-charge distribution near symmetry for thermal-neutron-induced fission of U235. *Physical Review C*, vol. 32, p. 184, 1985.

[28] X5-Team: *MCNP – A General Monte Carlo N-Particle Transport Code, Version 5, Volume I: Overview and Theory.* Los Alamos National Laboratory, USA, 2008.

[29] X5-Team: *MCNP – A General Monte Carlo N-Particle Transport Code, Version 5, Volume III: Developers Guide.* Los Alamos National Laboratory, USA, 2008.

Affiliations

Mikolaj Oettingen
AGH University of Science and Technology, Faculty of Energy and Fuels, Department of Nuclear Energy, Krakow, Poland, `moettin@agh.edu.pl`

Jerzy Cetnar
AGH University of Science and Technology, Faculty of Energy and Fuels, Department of Nuclear Energy, Krakow, Poland, `cetnar@mail.ftj.agh.edu.pl`

Tomasz Mirowski
The Mineral and Energy Economy Research Institute of the Polish Academy of Sciences, ul. Wybickiego 7, 31-261 Krakow, Poland

Received: 19.11.2014
Revised: 6.03.2015
Accepted: 7.03.2015