The Monte Carlo Method: Versatility Unbounded In A Dynamic Computing World
Chattanooga, Tennessee, April 17–21, 2005, on CD-ROM, American Nuclear Society, LaGrange Park, IL (2005)

CODE TDMCC FOR MONTE CARLO COMPUTATIONS OF SPATIAL REACTOR CORE KINETICS

A.K. Zhitnik, N.V. Ivanov, V.E. Marshalkin, S.P. Ognev, A.V. Pevnitsky, V.M. Povyshev, I.E. Ponomarev, V.I. Roslov, T.I. Semenova, V.A. Tarasov, V.P. Fomin
Russian Federal Nuclear Center – VNIIEF
37 Mir Ave., Sarov, Nizhni Novgorod region, Russia
vat@vniief.ru

T.A. Taiwo, W.S. Yang
Argonne National Laboratory
9700 S. Cass Avenue, Argonne IL, 60439
Taiwo@anl.gov

ABSTRACT

The Russian Federal Nuclear Center – VNIIEF has developed a Time-Dependent Monte Carlo Code (TDMCC) under an ISTC grant (ISTC Project #1086). The code is designed for Monte Carlo computations of time-dependent processes of VVER and PWR core neutron dynamics, both for emergencies and design transients. It can also be used to compute fuel element burnup and energy release fields in the core (steady-state conditions). The code employs direct simulations of neutron and nuclear kinetics in the core with detailed geometry and composition specifications and utilization of modern libraries of elementary neutron data. The code can be operated both independently and in conjunction with different thermal hydraulic codes. The applicability of the for analysis of VVER and PWR cores is demonstrated by assembly test problems.

Key words: Nuclear reactor, Dynamic processes, Time-dependent Monte Carlo method

1 INTRODUCTION

Currently, with the advent of modern computer systems (multiprocessor computers offering the possibility of parallel processing), it becomes more and more attractive to use Monte Carlo methods to analyze nuclear reactor cores. To date it has become possible to calculate a great many histories of neutrons, thereby ensuring a high statistical accuracy of core parameters of interest. The Monte Carlo methods are used mainly for time-independent quasi-static depletion calculations. For the Monte Carlo methods to become a tool for routine analysis of the reactor systems, they should contain features typically inherent in current tools using deterministic methods for core computations. These include the thermal-hydraulic feedback, efficient analysis of transients, computations of perturbations, processing of stochastic uncertainties in the burnup computations, and efficient algorithms for typical trade studies needed for development of optimized reactor designs.

The Russian Federal Nuclear Center – VNIIEF has developed a Time-Dependent Monte Carlo Code (TDMCC) [1] under an ISTC grant (ISTC Project #1086). The code is designed for computations of processes in VVER and PWR reactor cores both for design conditions and hypothetical transients. The code can also be used to compute fuel element burnup and energy release fields in the core (steady-state conditions) [2].

The code employs direct simulations of neutron and nuclear kinetics in the core:
- with the highest accuracy of detailed geometry and composition specifications
- using modern libraries of elementary neutron data.

Advantages (in terms of physics) of the Monte Carlo solutions to the kinetic equation are:
- absence of the group approximation (continuous neutron energy modeling);
- absence of the physical approximations inherent in the diffusion theory approach. Thus, the diffusion theory approximation is not quite applicable in the following cases: computations of "small" reactor cores and computations with partial core coolant voiding or local coolant boiling in the core;
- calculation of detailed characteristics on fuel elements and coolant;
- elementwise composition of each computational fragment.

Furthermore, there are neither "cells" nor "cell" group constants to be prepared in advance for a large number of options and cell states (in coolant temperature and density, in fuel temperature, in depletion, in state of neighbors, etc.), which results in faster preprocessing for the computations.

2 TDMCC ARRANGEMENT AND POSSIBILITIES

The code TDMCC has been developed on the basis of the C-95 code [3] (VNIIEF Monte Carlo code) designed for solution of the spectral kinetics equation of simultaneous neutron and gamma quanta transport.

The code can be executed both individually and in combination with different thermohydraulic codes (see, e.g., [4]).

The code is intended for computations on high-performance parallel architecture machines.

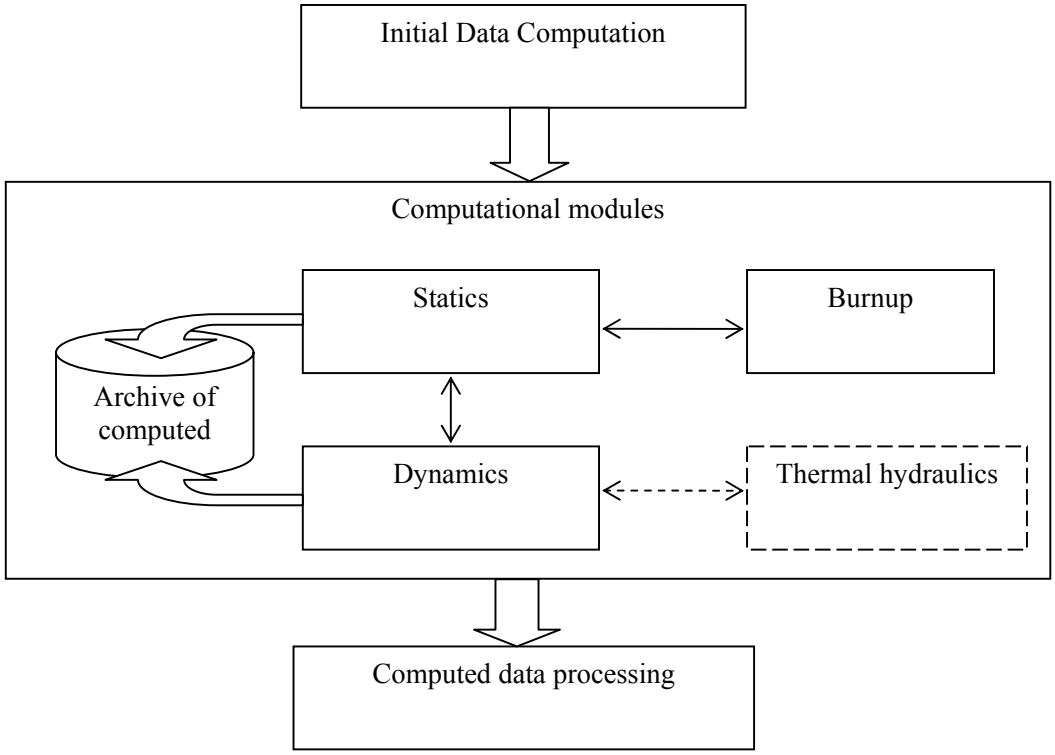
The initial data include the following information:

- problem geometry description;
- chemical composition of regions;
- neutron constants;
- computed data parameters;
- computational tactics;
- needed additional problem characteristics.

The initial data is set and processed with a stand-alone code.

The following libraries of neutron constants can be used: ENDF/B-V, ENDF/B-VI, ENDL-82, JENNDL3.2, CENDL , etc.

Geometry of systems. The code allows description of 3D-geometry systems. Geometry of systems is described using the building-block principle with a nested system. The block is described by a set of surfaces. The surfaces can be second-degree surfaces of revolution or planes. (Arbitrary second-degree surfaces are admitted, however in that case there is no program check of given data consistency). The nest system determines positions of the blocks relative to each other. Several nesting levels are, therefore, permitted. The nests of each next level indicate the position in the block, which is the basic system. If at some nesting level there is a block, in which there are next-level nests, then these nests are positioned together within the block.



Neutron-nuclear constants. The neutron transport computation involves, as a rule, neutron-nuclear constants of different versions of ENDL and ENDF/B libraries. (The code algorithm accounts for thermal Maxwellian nuclei motion during the process of cold cross section simulation.) For several elements there are interaction cross sections including chemical bonds [model $S(\alpha,\beta)$]. Note that the computations require neutron-nuclear interaction constants for varying medium temperature both in space and time.

Sources. The code offers a wealth of possibilities in setting the space and spectral-angular distribution of independent sources. Arbitrary surfaces and region systems as well as off-geometry items can serve as particle sources.

Results. The calculated functionals are subdivided into volumetric and surface ones. Each result can be distributed over different states of phase coordinates of the particle, which it is taken from. The surface results are currents and particle fluxes through system surfaces as well as their convolutions with arbitrary energy functions. (They are calculated in intersection of a given surface.) The volumetric results are fluxes and numbers of reactions. As a rule, the estimate in track length is used in their calculation. The estimate in collisions can also be used to calculate the number of reactions. The option to obtain cell group constants has been implemented for reactor computations.

Tools to enhance the computation efficiency. There is a significant stock of such tools in the code. These include: value over regions, splitting and roulette, exponential transformation, method of test particles (an analog of DXTRAN), local estimations, as well as different path cutting-off methods.

Additionally, in the TDMCC code, there is a large set of boundary conditions that allow problem solutions in lattice systems, both unbounded and with a finite number of cells.

3 A TECHNIQUE FOR INCLUSION OF THERMAL MOTION OF ATOMS

The program complex TDMCC employs a free-gas model for description of thermal motion of medium atoms. In the model, the distribution of n-kind atom velocities is of the form of Maxwellian distribution:

$$p_n(\vec{u}) = \left(\frac{A_n}{2\pi T(r)} \right)^{3/2} \cdot \exp\left(-\frac{A_n \vec{u}^2}{2T(r)} \right), \quad (1)$$

where A_n is the atom mass in the units of neutron mass; $T(r)$ is the medium temperature at point r. Two techniques for inclusion of the thermal motion are known.

One technique is used in deterministic methods. The technique is based on preliminary averaging of cold cross sections according to the Maxwell distribution followed by tabulation of the neutron energy and medium temperature averaged (thermal) cross sections. A disadvantage of the technique is the construction of additional tables and the need for interpolation between nodal temperatures, which introduces additional errors to the computations.

The other technique is used in Monte Carlo programs. It is based on cold cross sections and, hence, does not require construction of any additional tables. The averaging according to the Maxwellian distribution is achieved with neutron path modeling schemes using the method of "maximum cross section" [5].

Such a path modeling scheme has been used in the program complex C-95 for a few years. In the scheme, an important element is evaluation of maximum total cold cross section in energy range $[E_{L,} E_R]$, where

$$E_L = E\left(\max\left\{0, 1 - y_* \sqrt{T/(EA_n)}\right\}\right)^2, \; E_R = E\left(1 + y_* \sqrt{T/(EA_n)}\right)^2 \quad (2)$$

Here E is the current neutron energy, y_* is a parameter specifying the maximum difference of the relevant energy from the current one due to medium atom motion. Maximum thermal cross section is evaluated by averaging the maximum cross section and used to model collision points. The difference of the maximum cross section from the actual one is compensated by fictitious collisions, in which the neutron state remains unchanged. The experience gained suggests that there are two disadvantages in the scheme used. The number of the fictitious collisions can increase dramatically in problems where the neutron spectrum is concentrated in a low energy region or in a region of energies with resonance behaviors of cross sections.

The point is that in the low energy region the left limit of range is frequently zero, while the cold fission and capture cross sections behave like 1/v, where v is the neutron velocity modulus. Therefore, in this region the maximum cross section depends on the left limit of the range and can differ noticeably from the actual cross section on the right limit. Note that in these cases the collision frequency equal to the product of velocity by cross section is about constant. Hence, in the low energy region most efficient modeling schemes will be those based on maximum collision frequency.

In the resonance region, a few resonances get into the energy range $[E_{L,} E_R]$, so the maximum cross section can differ noticeably from the actual cross section. Ref. [6] suggests an idea to reduce the number of fictitious collisions through using thermal cross sections calculated

for some basic temperature T_0. The idea is that, even for the room temperature, the thermal cross sections behave smoother in the resonance region. Moreover, many neutron-material interaction processes are of threshold nature and the thresholds are in a high energy region, where the thermal motion effect becomes small. In the low and medium energy region, the principal processes are: elastic scattering, fission, and absorption. For absorption, the thermal motion effect is included completely in the cold absorption cross section averaging. In fission, the secondary particle distribution depends weakly on initial energy of neutron. Therefore the thermal motion effect is only important in simulations of secondary particles on elastic scattering.

Ref. [6] constructs a new scheme for modeling of neutron paths in medium of temperature T using the cross sections calculated for the basic temperature $T_0 \le T$. When modeling the secondary particle distribution, the thermal motion is included only on elastic scattering.

The scheme implementation requires a maximum number of basic cross sections in the energy range determined by the temperature difference $\Delta T = T - T_0$. For $T = T_0$ the fictitious collisions disappear completely. When modeling the elastic scattering, the medium atom velocity is represented as sum $\vec{u} = \vec{u}_\Delta + \vec{u}_0$, where velocity $\vec{u}_\Delta$ is determined by the basic cross sections and temperature difference ΔT and velocity $\vec{u}_0$ by the cold cross sections and temperature T_0. The second velocity is modeled with the method of rejections, through calculation of the maximum cross section in the energy range determined by temperature. The efficiency of the method of rejections is equal to the ratio of the cold elastic cross section to the maximum cross section.

Note that the implementation of the rejections is noticeably more cost-efficient than the fictitious collisions, so the new scheme is more efficient than the scheme based only on cold cross sections. However, the need to have the cold elastic cross section in addition to the thermal cross sections is a disadvantage of the new scheme. This disadvantage can be removed through transfer to an approximate simulation scheme, in which the maximum cold elastic cross section is set equal to the cross section in the current neutron energy. As a result, the number of the rejections is reduced to a minimum.

The following can be stated for justification of this approximation. On heavy elements the thermal motion effect is quite small, while on light elements the elastic scattering cross section behavior is smooth. Ref. [6] demonstrates on model problems that the approximate scheme accuracy is fairly good. It is therewith noted that thermal cross sections behave like 1/v in the low energy region. Hence, in the low energy region the modeling scheme with maximum collision frequency will be more efficient.

Schemes using cold and thermal cross sections have been implemented recently in the complex C-95. In the latter case the elastic scattering is modeled approximately. Either scheme automatically chooses a method based either on maximum frequency or maximum cross section.

4 DESCRIPTION OF THE TDMCC COMPUTATIONAL MODULE

The principal computational module of the code TDMCC is designed for model research computations of reactor core neutron dynamics. The computations include time dependent treatment for delayed neutron generation.

The basic distinction of TDMCC from C-95 is that a timestep-by-timestep computation is used rather than tracking the particle path from the generation point to loss from system

(absorption or leakage), as done in linear transport problems, or from fission to loss, as in computations of k_{eff}.

Ideologically, the algorithm used in the code is close to the one of ref. [7].

Principal features of the TDMCC modeling of neutron paths:

1. the delayed neutron source is given using the data from the preprocessing-step file

2. timestep-by-timestep computation is used

3. prompt and delayed neutrons are simulated separately

4. in the modeling of paths:

- the method of maximum cross sections is used
- thermal motion of medium nuclei is included

4.1 Modeling of prompt neutrons

The space-energy distribution of prompt neutrons is transferred to the timestep beginning from the previous step through an array containing information for an ensemble of N neutrons (N is given in the initial data and determines the problem statistics).

Ten numbers are assigned to each particles: block number, block exemplar number, region number, coordinates specifying the particle position in space (x, y, z), particle velocity vector direction cosines (cx, cy, cz), and reciprocal of particle velocity modulus.

This information describes the location and characteristics of each neutron in the ensemble and, hence, completely determines the system particle distribution.

The computation proceeds with a **constant** number of the ensemble particles **in timesteps** Δt. The data on the neutrons that were released at time t_0 and survived at time $t = t_0 + \Delta t$ is initial data in the modeling of the following step.

If the number of particles that survived at time T is larger than the ensemble size, then the data of the remaining particles is not recorded (but their weight is taken into account), otherwise the package is complemented to the original value by sampling from already available ones.

4.2 Modeling of delayed neutrons

Delayed neutrons are generated by nuclear fragments during fission reactions. The following equation describes a change in the fission fragments, which are delayed neutron sources:

$$\frac{dN}{dt} = A - \lambda \cdot N , \tag{3}$$

where N is the number of fission fragments.

The time of appearance of delayed neutrons relates to the half-lives of their fragment precursors. On the basis of this the delayed neutrons are typically divided into six groups.

The code C-95 (critical parameter computation) was used to compute numbers determining the nuclear fission point distribution over fuel elements and layers. The resultant numbers can be considered as components of 2D matrix A_{ij} of size $m \times n$, where m is the number of fuel elements $(m = 59)$, n is the number of layers, which the fuel elements have been divided into $(n = 10)$. A three-dimensional array can be readily obtained from it that reflects the distribution of the delayed neutrons from the steady-state conditions over fuel elements, layers, and time exponents. $A_{ijk} = \dfrac{a_{ij} \cdot p_k}{\lambda_k}$. However, this array only gives the distribution of delayed neutrons in

the system and not their quantity. The coefficient relating the number of delayed neutrons and the number of their precursors is found from the condition that the delayed neutron source does not change with time.

Delayed neutron weight per prompt neutron for modeled timestep depends on the timestep length. Therefore the delayed neutron release frequency is varied depending on the timestep length during the ensemble modeling.

The execution of the TDMCC code results in the step-by-step computation of system neutron amount normalized to one initial (T=0) neutron.

The capability of control rod motion modeling is embedded in the developed computational module. The magnitude of the change in the position of the rod relative to its original level can be

- either given in the initial data of the problem by specification of the number of the package, after which the rod position should be changed,
- or typed from the keyboard during the computation.

4.3 Parallelization algorithm

The programs were parallelized using a parallel algorithm for computing path groups within a package, which had been developed previously in the complex C-95.

Its idea is that the parallelization of processes is executed within a single package composed of M paths in groups m paths in each.

All the processes, except for the zeroth one, simulate the paths, while the zeroth process is a control process.

The data on delayed neutrons is stored on the zeroth process. The initial parameters of the delayed neutron path are computed by the zeroth process and transferred for further computation to the first process which is free at that time.

At the package computation end the zeroth process sequentially receives the phase parameters of the neutrons that survived until the timestep end, which have been obtained by the computational process on the next path group, and arrays containing information about the delayed neutron generation on the path being modeled. On the package modeling completion the zeroth package renormalizes the ensemble of the particles that survived until the timestep end to the initial package for the next step and calculates the number of the delayed neutrons that have been generated during the timestep. Then everything is repeated until the computation completion.

The efficiency of this parallelization method for the problem under consideration is no less than 90%.

5 COMPUTATION OF TRANSIENT FOR A MODEL FUEL ASSEMBLY

Using a model fuel assembly typical of PWR and VVER type reactors as an example, the problem of computing reactor runaway from the steady state in prompt injection of positive reactivity $\approx 0.5\beta$ was considered. This reduced geometry problem was selected as a proof of the computational capability and, additionally, to minimize computational cost and time.

5.1 Setting up the computation

The model fuel assembly is designed as a set of hollow zirconium tubes 10.92 mm in outer diameter and 0.66 mm in wall thickness and one continuous metal rod of the same diameter.

Inside each zirconium tube is UO_2 nuclear fuel of 10.2 g/cm^3 density. The assembly height is H =3.7 m. The number of the rods is 204. The distance between the fuel rod centers is 14.4mm. In the fuel assembly there are 4 longitudinal channels to insert control rods and one central guide channel for installation of the fuel assembly in the reactor that contains a 10.92-mm-diameter steel rod. The 14-mm-diameter control rod is located inside a 1-mm-thick steel cylindrical shell. The rod is inserted into the fuel assembly as deep as one fourth of the assembly height. From above and from below the fuel element zone was covered with steel lids with water interlayer.

The fuel assembly is cooled by liquid flowing along the rods.

By virtue of symmetry of the fuel assembly, ¼ of the structure is considered and specular boundary conditions are introduced on the planes of symmetry. The overall dimensions of the computational domain are 108x108x3700 mm.

The computational geometry is presented in Figures 1, 2.

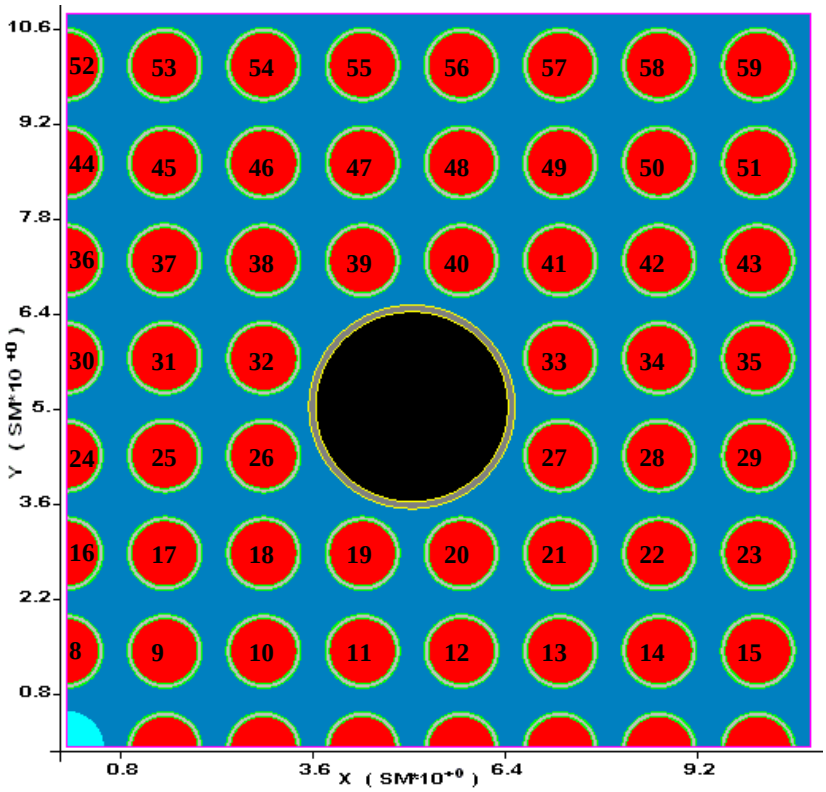


Figure 1. Geometry of the TDMCC computation. (Top view)

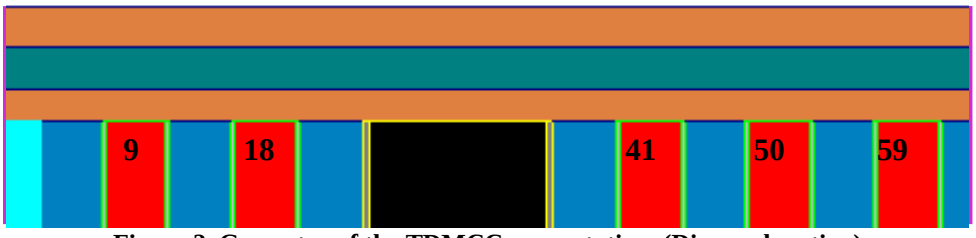


Figure 2. Geometry of the TDMCC computation. (Diagonal section)

Table 1 Compositions of the regions corresponding to the regions of the same colors in Figures 1, 2.

Region:	**Composition** (10^{24} nuclei)
- Borated water (for ρ_0=1 g/cm^3)	**H** (6.672E-02), **O** (3.336E-02), **B10** (4.270E-05)
- Fuel element rod	**U-235** (8.283E-04), **U-238** (2.193E-02),**O** (4.551E-02)
- Fuel cladding	**Zr (**4.298E-02**)**
- Steel rod	**Fe (**8.465E-02**)**
- Control rod	**B10** (1.959E-02), **B11** (8.925E-02), **C (**2.721E-02**)**
- Control rod shell	**Fe (**8.465E-02**)**
- Lid	**Fe** (8.465E-02)
- Water in the lid	**H (**6.672E-02**), O (**3.336-02**)**
- Reflecting boundary	- // - // - // - // -

In the neutron kinetics simulations, nuclear thermal motion is included with account for chemical bonds – a so-called *S(α,β)* model. The simulations including the chemical bonds use interaction cross sections and energy-angle distribution corresponding to a given temperature.
The computations involved constants produced using the ENDF/B-V library. In the neutron-water interaction description, the model *S(α,β)* (hydrogen-in-water) was used for hydrogen at 600K temperature.

5.2 Results of the dynamic test computation

At the initial time the control rod was inserted by half the assembly height into the fuel assembly. Concentration of boron B^{10}) was selected so that the system is at steady state (k_{eff} = 1.000891 (0.007%)).

The dynamic computation consisted of two steps.

Step 1 corresponds to establishment of steady-state conditions in the model fuel assembly under consideration. For this purpose 120 timesteps of 0.0005sec duration were computed. According to the previously described technique for Monte Carlo dynamic reactor computation, the computation was conducted with a constant number of particles in the ensemble. For that computation this is 100000 neutrons. The computed data presented in Figure 3 suggests that the system had tapered off to a steady-state regime. In what follows, in all the figures, the ordinates are the system neutron quantities normalized to one initial neutron.

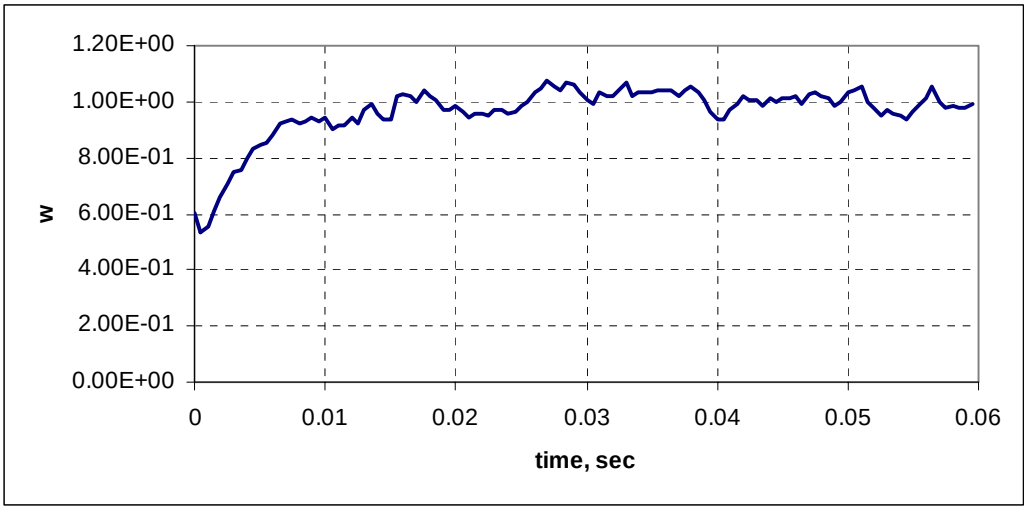


Figure 3. Step 1. Establishment of steady-state conditions

The beginning of computational step 2 corresponds to the time of the prompt displacement of the rod to a height of 30 cm, which leads to an increase in the neutron effective multiplication factor to k_{eff}= 1.0032700891 (0.0107%), i.e. the system reactivity is about 0.5 β .

For step 2, 7690 timesteps of 0.0005sec duration were computed, which makes 3.846 seconds for the system under consideration.

Results of the Step 2 computations are presented in Figure 4.

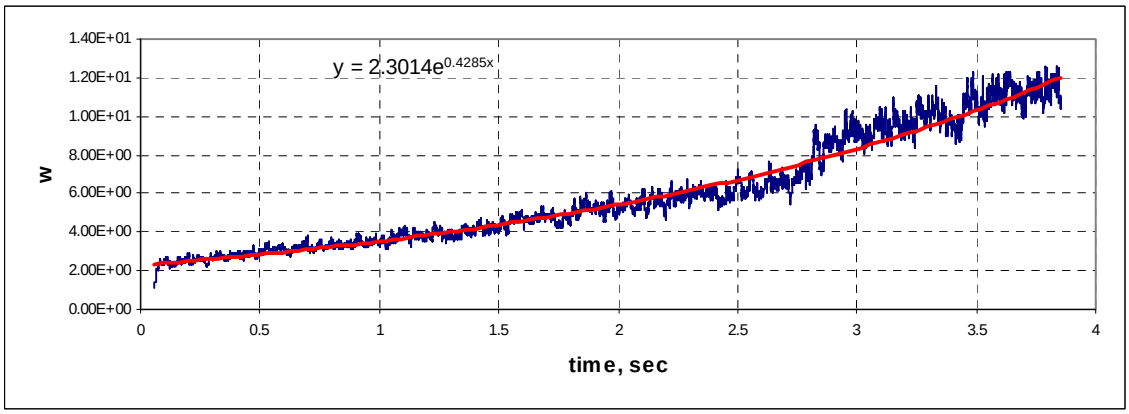


Figure 4. Step 2. Time history of the system on its reactivity increase

That computation was performed by the parallel version of the dynamics program, as in the one-processor mode the computation of one timestep on PIII-1133MHz takes 3 hours of computer time.

The computations were conducted at the Computer Center of Sarov Laboratories.

The total computer time of the computation is about 26400 hours, the actual time is 910 hours, or ~38 days. (6 nodes, one processor PIV-2800MHz in each).

Figure 5 plots the computed time history of the system on the control rod removal in uniprocessor and multiprocessor versions.

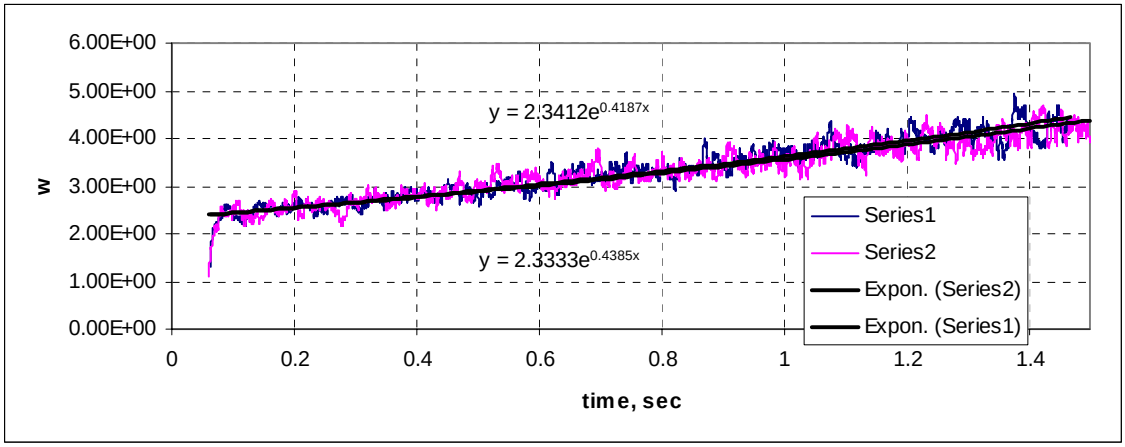


Figure 5. Comparison between the uniprocessor (Series1) and multiprocessor (Series2) versions of the computations

6 CONCLUSIONS

Code TDMCC for Monte Carlo computations of time-dependent processes has been developed.

The obtained results suggest that the code can be employed for computations of spatial kinetics of VVER and PWR reactor cores.

7 ACKNOWLEDGEMENTS

The work was carried out under an ISTC Grant (ISTC Project #1086).

8 REFERENCES

1. A.K. Zhitnik, N.V. Ivanov, V.E. Marshalkin, S.P. Ognev, V.M. Povyshev, I.E. Ponomarev, V.I. Roslov, T.I. Semenova, V.A. Tarasov, V.P. Fomin, T. A. Taiwo and W. S. Yang, "The TDMCC Monte Carlo Capability for Spatial Kinetics Calculations of Reactor Cores", *Trans. Am. Nucl. Soc*., **91**, 248 (2004).

2. E.V. Artemyeva, V.V. Bakanov, A.K. Zhitnik, V.E. Marshalkin, V.M. Povyshev, I.E. Ponomarev, A.B. Ronzhin, V.P. Fomin, and V.A. Tarasov, "**A** package for computing start isotope burnup and actinide and fission product recovery for code TDMCC", *The Monte Carlo Method: Versatility Unbounded In A Dynamic Computing World*, Chattanooga, Tennessee, April 17–21, 2005, on CD-ROM.

3. E.N. Donskoy, V.A. Yeltsov, A.K. Zhitnik et al., "Monte Carlo Method at VNIIEF", *VANT, Ser.:Math.Model.Phys.Proces.*, Is.2, pp.61-64, 1993.

4. V. V. Bakanov, A. K. Zhitnik, V. N. Motlokhov, S. P. Ognev, V. I. Romanov, A. A. Ryabov, V. A. Tarasov, S. A. Tareev, V. P. Fomin, D. P. Weber, T. A. Taiwo and W. S. Yang,

"TDMCC Monte-Carlo Package Coupled with STAR-CD Thermal-hydraulics Code ", *Trans. Am. Nucl. Soc*., **91**, 250 (2004).

5. I.M. Sobol, *Numerical Monte Carlo methods*, Nauka Publishers, Moscow, 1973.
6. A.N. Ivanov and N.V. Ivanov, "Account for thermal motion of medium atoms when solving neutron transport problems with the Monte Carlo method", *International Seminar Super-Computing and Computer Simulation*, Sarov, 2003.
7. J. A. Fleck and J.D.Cummings, "An implicit Monte Carlo Scheme for Calculating Time and Frequency Dependent Nonlinear Radiation Transport", *Journal of Computational Physics*, **V. 8**, p. 313-342 (1971).