A Monte Carlo Transient Multilevel Implementation Applied to the C5G7-TD3 Benchmark

Evan S. Gonzalez and Brian C. Kiedrowski

Department of Nuclear Engineering and Radiological Sciences, University of Michigan
2355 Bonisteel Blvd, Ann Arbor, MI 48109

esgonz@umich.edu, bckiedro@umich.edu

ABSTRACT

The transient multilevel (TML) method, multilevel time-stepping scheme using the predictor-corrector quasi-static method (PCQM) has been implemented in the Shift Monte Carlo particle transport solver. The Monte Carlo shape function is solved via transient fission source iteration, while the amplitude function is solved using a transient Coarse Mesh Finite Difference (CMFD) solver, and the CMFD flux is further factorized into shape and amplitude functions, where the CMFD amplitude function is now solved via the exact point kinetics equations (EPKEs). The implementation was tested using the C5G7-TD3 multigroup reactor transient benchmark. The TML method implementation results agree with the same simulation performed in MPACT, with L_∞, L_2, and L_1 norms all less than 4%. Results show that the TML method allowed larger Monte Carlo time steps to be taken throughout the transient when compared to the standard PCQM implementation, yielding about an order of magnitude performance improvement in simulation time.

1. Introduction

The last two decades have seen an increased demand in the reactor physics community for transient simulation capabilities that use neutron transport physics to support nuclear reactor design analyses. Until recently, these methods have largely been confined to deterministic transport methods [1–3] despite their dependence on approximations to energy-dependent nuclear data and complicated geometries. While the Monte Carlo (MC) method does not rely on these approximations, its application to transient calculations has been limited by its high computational expense. As computing power continues has continued to advance, MC has become a more tractable option and several transient MC implementations have begun to appear in the neutron transport literature.

Sjenitzer and Hoogenboom developed the dynamic MC method, which stochastically samples the birth and decay of delayed neutron precursors throughout a transient that has proven to be very accurate, but too expensive, for practical use [4]. Hackemack et al. implemented a quasi-static method in a fixed-source MC code that allowed larger time steps by performing fine-time resolution calculations with a point kinetics solver but suffered from individual histories not terminating for positive reactivity insertions [5]. Jo et al. addressed this issue by solving the MC transport problem with fission source iteration to guarantee the termination of individual histories [6]. The work described in this summary presents the latest developments in the transient benchmark results generated on a research development branch of Shift, an MC code developed at Oak Ridge National Laboratory (ORNL) [7].

2. Shift Transient Method

Shift's standard k-eigenvalue routine has been modified to support transient calculations. The time-integrated k-eigenvalue approximation to the neutron transport equation in operator form with a backwards Euler time discretization is

$$\left(\boldsymbol{M}+\frac{1}{v\left(E\right)\Delta t_n}\right)\psi^{l+1}\left(\boldsymbol{x},\hat{\boldsymbol{\Omega}},E,t_n\right) = \underbrace{\frac{\psi\left(\boldsymbol{x},\hat{\boldsymbol{\Omega}},E,t_{n-1}\right)}{v\left(E\right)\Delta t_n}}_{①} + \underbrace{\boldsymbol{F}_{\mathrm{p}}\psi^{l}\left(\boldsymbol{x},\hat{\boldsymbol{\Omega}},E,t_n\right)}_{②} + \sum_{j=1}^{J}\underbrace{\boldsymbol{S}_{\mathrm{d},j}\psi\left(\boldsymbol{x},\hat{\boldsymbol{\Omega}},E,t_{n-1}\right)}_{③} f_{1,j} + \sum_{j=1}^{J}\underbrace{\boldsymbol{F}_{\mathrm{d},j}\psi\left(\boldsymbol{x},\hat{\boldsymbol{\Omega}},E,t_{n-1}\right)}_{④} f_{2,j} + \sum_{j=1}^{J}\underbrace{\boldsymbol{F}_{\mathrm{d},j}\psi^{l}\left(\boldsymbol{x},\hat{\boldsymbol{\Omega}},E,t_n\right)}_{⑤} f_{3,j} \quad (1)$$

where $\boldsymbol{M}$ is the transport operator, $\boldsymbol{F}_{\mathrm{p}}$ and $\boldsymbol{F}_{\mathrm{d}}$ are the prompt and delayed fission operators, respectively, $\boldsymbol{S}_{\mathrm{d}}$ is the accumulated delayed neutron source operator, ψ is the neutron angular flux, v is velocity, $\boldsymbol{x}$ is position, $\hat{\boldsymbol{\Omega}}$ is direction, E is energy, and t is time. The indices are defined such that l is the fission source iteration index, j is the delayed neutron precursor group, and n is the time step index. The time-interpolated weighting factors for the delayed fission source terms appear as a consequence of approximating the fission source terms with linear time-dependence and are

$$f_{1,j} = e^{-\lambda_j \Delta t_n}\,, \quad (2a)$$

$$f_{2,j} = \frac{1-e^{-\lambda_j\Delta t_n}-\lambda_j\Delta t_n e^{-\lambda_j\Delta t_n}}{\lambda_j\Delta t_n}\,, \quad (2b)$$

$$f_{3,j} = \frac{e^{-\lambda_j\Delta t_n}\left(1-e^{\lambda_j\Delta t_n}+\lambda_j\Delta t_n e^{\lambda_j\Delta t_n}\right)}{\lambda_j\Delta t_n}\,. \quad (2c)$$

The left-hand side of Eq. (1) is implemented by augmenting the total cross section at the beginning of a random walk by the factor $\frac{1}{v\Delta t}$. This augmentation accounts for the sampling of an additional "time census" event when performing the distance-to-collision calculation.

The implementation of the source terms on the right-hand side of Eq. (1) uses two separate particle source banks: one passed between fission source iterations (terms ② and ⑤), and one passed between time steps (terms ①, ③, and ④). To decrease computational time, all five source terms are sampled implicitly at all collisions by storing a copy of the incident particle to its respective source bank with an appropriate weighting factor. The individual source terms and their corresponding weight modifications upon collision are:

① Neutrons from time t_{n-1} that were still in flight by the time t_n. These "time census" neutrons are implicitly sampled by augmenting the total cross section and require a weight modification of

$$w_{\text{banked}} = w_{\text{incident}}\left(\frac{\frac{1}{v(E)\Delta t_n}}{\Sigma_{\mathrm{t}}^{n}\left(\boldsymbol{x},E\right)+\frac{1}{v(E)\Delta t_n}}\right)\,. \quad (3a)$$

Since the distance-to-collision calculation at the start of the random walk was modified, to preserve the

mean response, the particles exiting a collision must also have their weight modified as

$$w_{\text{collision}} = w_{\text{incident}} \left(\frac{\Sigma_{\text{t}}^{n}(\boldsymbol{x}, E)}{\Sigma_{\text{t}}^{n}(\boldsymbol{x}, E) + \frac{1}{v(E)\Delta t_n}} \right) . \tag{3b}$$

(2) Neutrons resulting from prompt fission. This bank is already handled in steady-state k-eigenvalue calculations and therefore required no modification to the solver.

(4) Neutrons resulting from delayed neutron precursors that were accumulated over the time step Δt_n and numerically weighted as being sourced from the previous time step, t_{n-1}, with a weight modification of

$$w_{\text{banked}} = w_{\text{incident}} f_{2,j} . \tag{3c}$$

(5) Neutrons resulting from delayed neutron precursors that were accumulated over the time step Δt_n and numerically weighted as being sourced from the current time step, t_n, with a weight modification of

$$w_{\text{banked}} = w_{\text{incident}} f_{3,j} . \tag{3d}$$

A simplified branching process that neglects energy-dependence is shown in Fig. 1 to depict the implicit banking of source terms at each collision site.

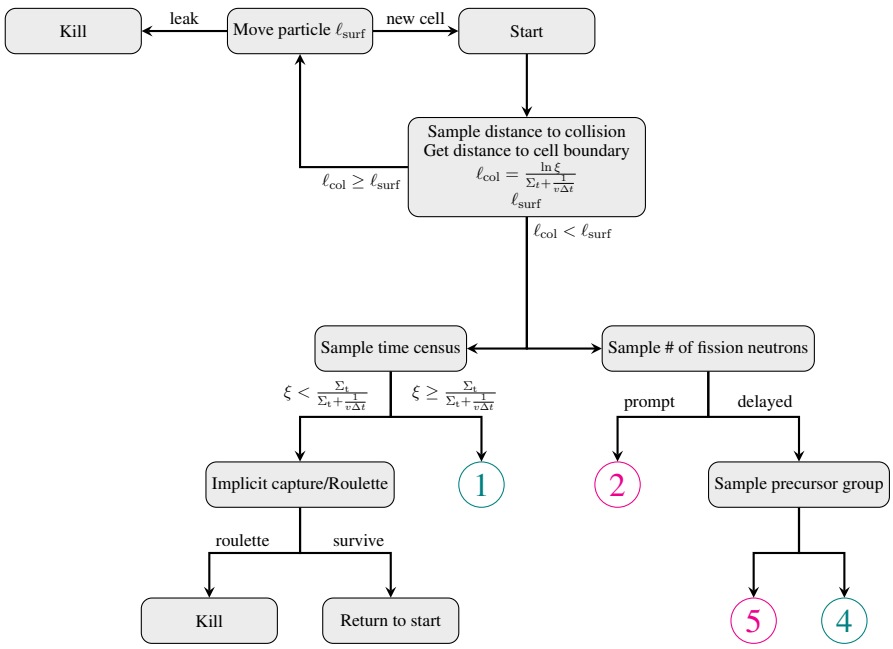

Figure 1: Simplified neutron history branching in transient fission source iteration scheme.

Source term (3) is a special term that accounts for the accumulation of delayed neutrons from the previous time step. Copies of the particles in the time step iteration bank at time step t_{n-1} are pushed to the time step iteration bank at t_n with weight modification

$$w_{t_n} = w_{t_{n-1}} f_{1,j} . \tag{4}$$

A particle comb is performed in between time steps to prevent exponential growth/decay of the time step iteration bank. Figure 2 depicts the transfer of source terms between fission source iterations and time

steps. Fission source terms (2 and 5) are passed between fission source iterations. During the active fission cycles, time step source terms (1 and 4) are tallied and read in as source terms at the start of the next time step. Finally, the accumulated delayed neutron term (3) is read from the previous time step iteration bank at t_{n-1}.

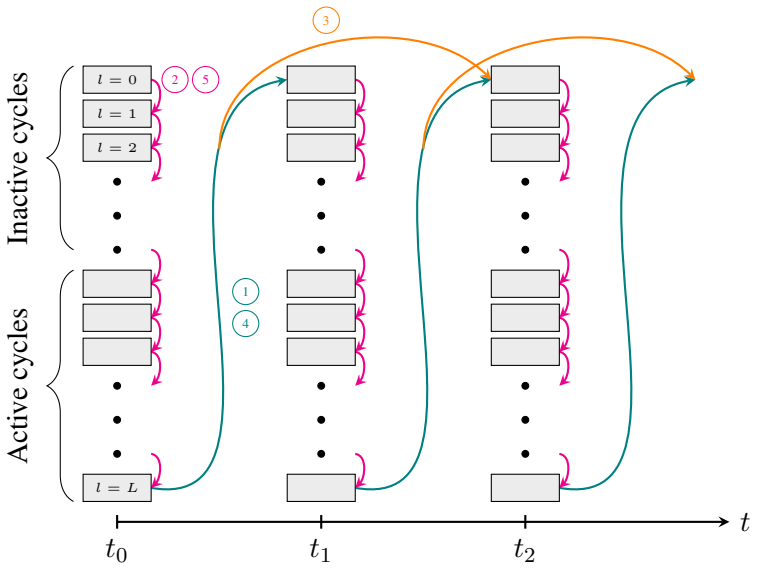

Figure 2: Transfer of source terms between fission source iterations and time steps.

2.1. Transient Multi-Level Method

Solving the time-dependent transport problem explicitly with MC can be prohibitively expensive because very small time steps are required to accurately capture the accumulation of the delayed neutron source. Quasi-static methods [8] address this problem by factorizing the neutron flux (scalar or angular) into the product of a shape function $\varphi(\boldsymbol{x}, \hat{\boldsymbol{\Omega}}, E, t)$ that is loosely dependent on time, and an amplitude function $p\,(t)$ that is dependent exclusively on time and computationally inexpensive relative to the shape function. This factorization relies heavily on the assumption that the shape function varies much more slowly in time than the amplitude function and can be updated much less frequently. The predictor-corrector quasi-static method (PCQM) [9,10] is the used in this work; the general steps are detailed as follows:

1. Evaluate the forward and adjoint steady-state transport fluxes (ψ_0 and $\psi_0^\dagger$, respectively). Compute a normalization constant $c = \left\langle \psi_0^\dagger, \frac{1}{v}\psi_0 \right\rangle$ to ensure uniqueness of the shape function.
2. Solve the transient transport problem on the coarse time scale to generate the "predictor" flux ψ^{p}. Evaluate the shape function as $\varphi = \frac{c}{\left\langle \psi_0^\dagger, \frac{1}{v}\psi^{\mathrm{p}} \right\rangle}\psi^{\mathrm{p}}$.
3. Solve the amplitude function p on fine time steps.
4. Use the amplitude function to "correct" the predictor flux $\psi^{\mathrm{c}} = \varphi p$.

The Transient Multilevel (TML) method [11,1] adds a nested time discretization layer to the PCQM (see Fig. 3), where the Coarse Mesh Finite Difference (CMFD) equations [12] are now used to compute a space

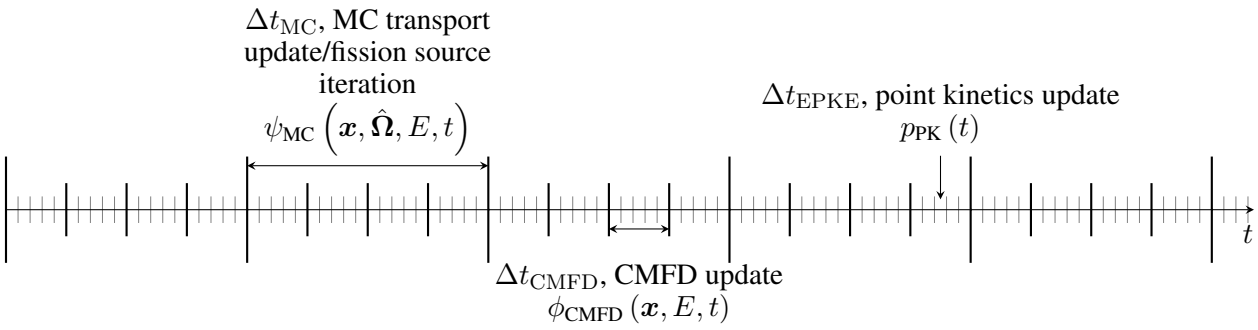

Figure 3: Transient Multilevel time discretization scheme.

and energy-dependent amplitude function for the factorized MC flux. The CMFD parameters (multigroup cross sections and diffusion coefficients) are derived directly from MC tallies on the macro time scale (Δt_{MC}) and are interpolated at the CMFD time steps Δt_{CMFD}. The CMFD scalar flux is used as an amplitude weighting function for particles sourced from the time step iteration bank in the MC simulation. The TML method also uses the solution of the exact point kinetics equations (EPKEs) [13] as a global amplitude function for the factorized CMFD scalar flux. All point kinetics parameters (except for the reactivity) in the Shift implementation are derived from MC tallies and interpolated at the EPKE time steps (Δt_{EPKE}). The reactivity is derived from an adjoint weighting of the CMFD fission and transport operators (see Hou [14] for the exact definition used). The adjoint weighting function is currently the solution from an SP_3 adjoint eigenvalue solver that is implemented (but not advertised) in Denovo [15] until native MC adjoint-weighted tallies are implemented.

2.2. Transient Coarse Mesh Finite Difference Equations

The transport-corrected multigroup diffusion approximation to Eq. (1) integrated over a single CMFD spatial element and normalized by its volume is

$$\begin{aligned}
\sum_{u\in x,y,z} \frac{1}{\Delta_l^u} \Big[& \left(-\tilde{D}^{n,u,g}_{l-\frac{1}{2},m,n} - \hat{D}^{n,u,g}_{l-\frac{1}{2},m,n}\right) \overline{\overline{\phi}}^{n,g}_{l-1,m,n} \\
& + \left(\tilde{D}^{n,u,g}_{l-\frac{1}{2},m,n} + \tilde{D}^{n,u,g}_{l+\frac{1}{2},m,n} - \hat{D}^{n,u,g}_{l-\frac{1}{2},m,n} + \hat{D}^{n,u,g}_{l+\frac{1}{2},m,n}\right) \overline{\overline{\phi}}^{n,g}_{l,m,n} \\
& + \left(-\tilde{D}^{n,u,g}_{l+\frac{1}{2},m,n} + \hat{D}^{n,u,g}_{l+\frac{1}{2},m,n}\right) \overline{\overline{\phi}}^{n,g}_{l+1,m,n} \Big] + \overline{\overline{\Sigma}}^{n,g}_{\mathrm{t}_{l,m,n}} \overline{\overline{\phi}}^{n,g}_{l,m,n} \\
& = \sum_{h=1}^{G} \overline{\overline{\nu_{\mathrm{s}}\Sigma}}^{n,h\to g}_{\mathrm{s}_{l,m,n}} \overline{\overline{\phi}}^{n,h}_{l,m,n} + \overline{\overline{\chi}}^{n,g}_{l,m,n} \overline{\overline{S}}^{n}_{\mathrm{F}_{l,m,n}} + \overline{\overline{S}}^{n,g}_{\mathrm{tr}_{l,m,n}} ,
\end{aligned} \tag{5}$$

where the fission source and transient source terms are defined, respectively, as

$$\overline{\overline{S}}^{n}_{\mathrm{F}_{l,m,n}} = \frac{1}{k_{\mathrm{eff}}} \sum_{h=1}^{G} \overline{\overline{\nu_{\mathrm{f}}\Sigma}}^{n,h}_{\mathrm{f}_{l,m,n}} \overline{\overline{\phi}}^{n,h}_{l,m,n} , \tag{6a}$$

$$\overline{\overline{S}}^{n,g}_{\mathrm{tr}_{l,m,n}} = \overline{\overline{A}}^{n,g}_{l,m,n} \overline{\overline{\phi}}^{n,g}_{l,m,n} + \overline{\overline{B}}^{n,g}_{l,m,n} \overline{\overline{S}}^{n}_{\mathrm{F}_{l,m,n}} + \overline{\overline{C}}^{n,g}_{l,m,n} . \tag{6b}$$

In Eq. (5) $\hat{D}$ and $\tilde{D}$ are specially-defined diffusion coefficients that relate the scalar flux to the net current and make the CMFD approximation consistent with higher-order transport solutions. The indices are defined such that superscript n corresponds to time step index, h and g correspond to incoming and outgoing

energy groups, repectively, subscript l, m, n correspond to spatial cells, and u, v, w can correspond to any spatial dimensions x, y, z relative to the selection of u. The Δ_l^u values correspond to the side length of a spatial cell. Quantities denoted with a double overbar $\bar{\bar{\cdot}}$ are volume-averaged and those denoted with a single overbar $\bar{\cdot}$ are surface-averaged quantities; both of which are derived directly from MC tallies. This notation and the definition of the diffusion coefficients are thoroughly explained in the OpenMC documentation [16]. The transient source coefficents A, B, and C, seen in Eq. (6b) are derived by Zhu in [1]. The transient formulation of the CMFD equations can be solved with any standard linear algebra solver.

3. C5G7-TD3 Transient Benchmark

The C5G7-TD benchmark simulates various time-dependent reactivity insertions applied to a small light water reactor with eight UO_2 assemblies and eight mixed oxide (MOX) assemblies. The benchmark specifications provide multigroup cross sections for all materials as well as delayed neutron precursor group decay fractions, decay constants, and fission spectra. More specific details about the C5G7-TD reactor can be found in the benchmark specification [14]. Exercise TD3 simulates four transient events by linearly varying the core moderator density for a 2D version of the reactor geometry with the time-dependent profile seen in Fig. 4.

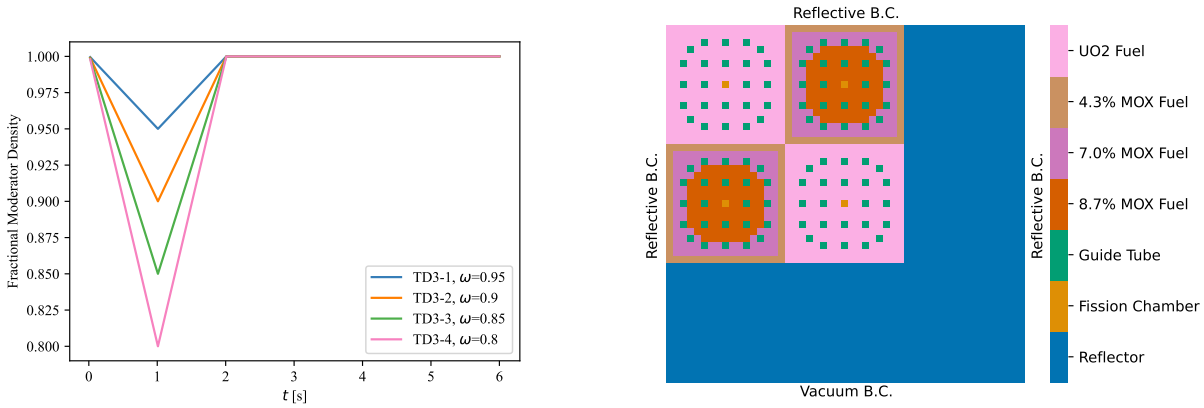

Figure 4: Core average moderator density change in TD3 exercises (left) and C5G7 quarter-core configuration (right).

The TD3 simulation results from Shift have been compared with the same transient simulation with MPACT, a 2D/1D deterministic code maintained by the University of Michigan and ORNL that uses Method of Characteristics in the radial direction and an SP_3 nodal method in the axial direction [17]. A detailed description of the C5G7-TD benchmark results with MPACT is provided in [3]. The TD3 problem has also been simulated by Shaner [18] using a frequency transform method [19] implemented in OpenMC, but has not been directly compared with Shift in this summary.

4. Results

A PCQM implementation (without the CMFD intermediate layer) was first implemented in Shift to test the transient fission source iteration implementation and provide a benchmark to asses performance improvements upon implementing the CMFD layer. The PCQM implementation is discussed in [20] and the results are shown in Fig. 5. The results show that stochastic noise in the dynamic reactivity after $t > 2\,\mathrm{s}$ is amplified by the EPKE solver, a phenomenon also observed and discussed by Shaner [18]. Figure 6 presents the same simulations with stochastic noised in the dynamic reactivity removed for $t > 2\,\mathrm{s}$, i.e. $\rho\,(t > 2\,\mathrm{s}) = 0$, which smooths the resulting fractional core fission rate profile.

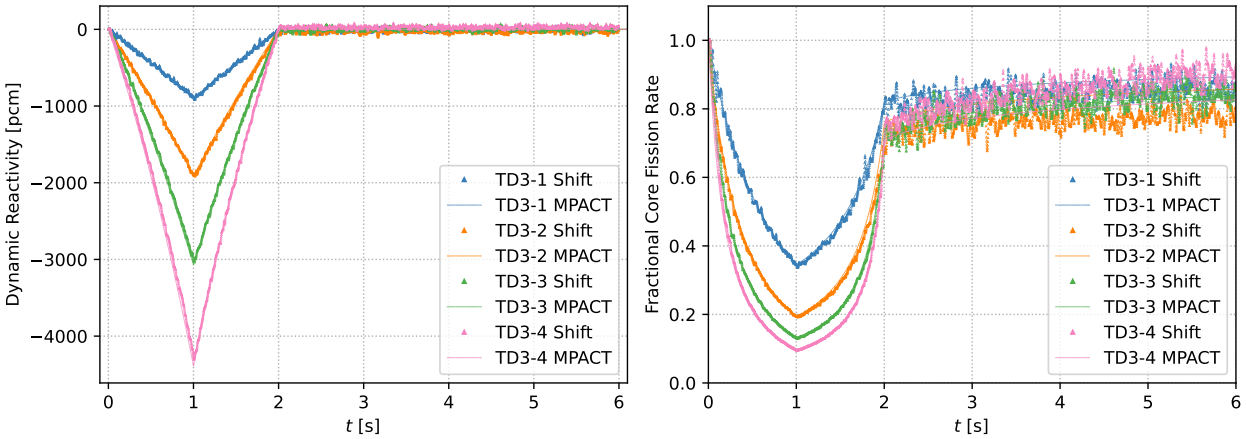

Figure 5: Dynamic reactivity and relative fission rate of the PCQM implementation.

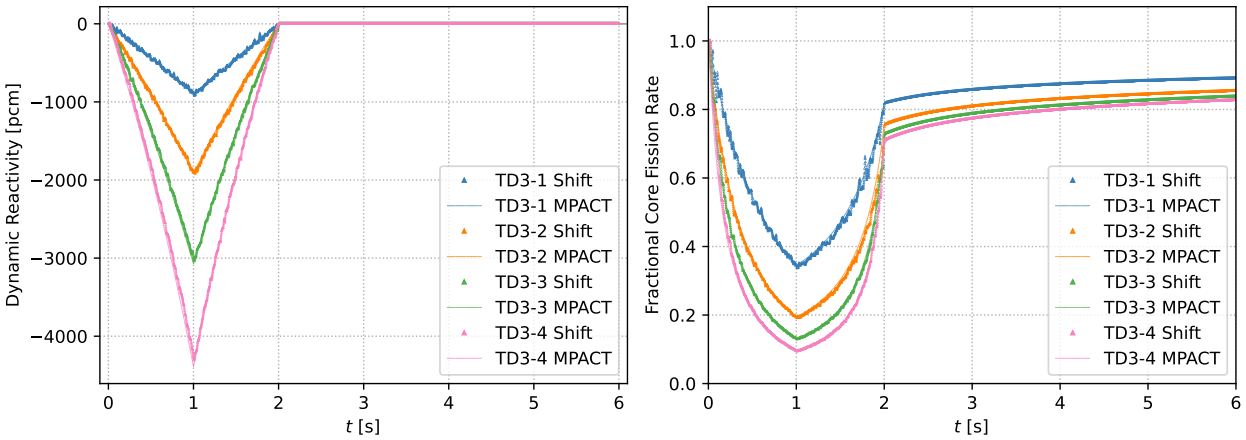

Figure 6: Dynamic reactivity and relative fission rate of the PCQM implementation with stochastic noise removed for $t > 2\,\mathrm{s}$.

Figure 7 shows the results of the TD3 simulations using the TML implementation, now with the CMFD layer. The TML method smooths out stochastic noise in both the dynamic reactivity and corresponding fractional core fission rate. This is likely because MC tallies (and their accompanying stochastic noise) are not directly driving the EPKE solution. Instead, EPKE reactivities are from the CMFD solutions that use linearly-interpolated cross section parameters. Several norm comparisons are given in Table I to asses TML performance improvements over the PCQM implementation. The norm evaluations decrease for almost every single TD3 case regardless of norm specification. In particular maximum relative difference between any given time step evaluation (L_∞ norm) shows almost an order of magnitude improvement.

The PCQM simulations used an MC time step of $\Delta t_{\mathrm{MC}} = 0.1\,\mathrm{s}$ and an EPKE time step of $\Delta t_{\mathrm{EPKE}} = 0.002\,\mathrm{s}$. While a rigorous convergence analysis was not performed to determine these time step selections, several iterations of user visual inspection were performed to maximize the allowable MC time step while matching the MPACT amplitude shape. Following this same procedure, the TML simulations were executed using an MC time step of $\Delta t_{\mathrm{MC}} = 0.25\,\mathrm{s}$, a CMFD time step of $\Delta t_{\mathrm{CMFD}} = 0.05\,\mathrm{s}$, and an EPKE time step of $\Delta t_{\mathrm{EPKE}} = 0.002\,\mathrm{s}$. Ideally, the use of a larger MC time step would bring down the total

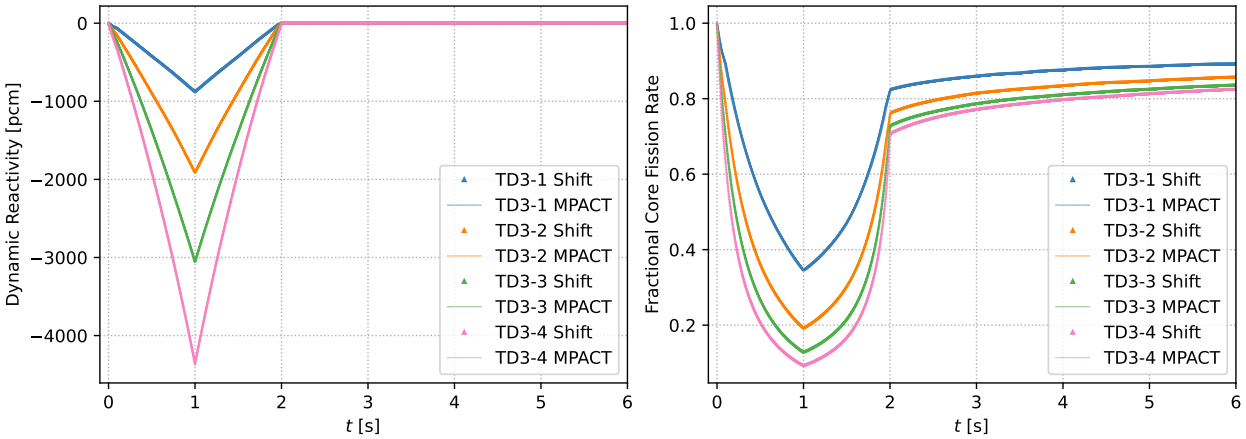

Figure 7: Dynamic reactivity and relative fission rate of the TML implementation with stochastic noise removed for $t > 2\,\mathrm{s}$.

Table I: Norm comparisons to MPACT solution for both PCQM and TML implementations.

Norm		TD3-1	TD3-2	TD3-3	TD3-4
$L_\infty = \max_n \left[\frac{p_{n,\mathrm{Shift}} - p_{n,\mathrm{MPACT}}}{p_{n,\mathrm{MPACT}}}\right]$	PCQM	8.42%	13.48%	10.12%	12.38%
	TML	2.40%	3.65%	2.83%	2.23%
$L_2 = \frac{1}{N}\left[\sum_n \left(\frac{p_{n,\mathrm{Shift}} - p_{n,\mathrm{MPACT}}}{p_{n,\mathrm{MPACT}}}\right)^2\right]^{\frac{1}{2}}$	PCQM	0.28%	0.47%	0.28%	0.46%
	TML	0.09%	0.12%	0.13%	0.13%
$L_1 = \frac{1}{N}\sum_n \frac{p_{n,\mathrm{Shift}} - p_{n,\mathrm{MPACT}}}{p_{n,\mathrm{MPACT}}}$	PCQM	-1.51%	-3.60%	-0.47%	2.57%
	TML	0.57%	-0.10%	-0.71%	-1.04%

simulation time as the MC solve is much costlier than both the CMFD and EPKE solves. A rough timing comparison is shown in Table (II).

Table II: CPU time comparison between the TD3 solution methods.

TD3 Case	Shift-PCQM (6 s)	Shift-TML (6 s)	MPACT (10 s)
1	2510.1 h	395.0 h	30.6 h
2	3067.1 h	395.1 h	29.7 h
3	2533.1 h	398.0 h	29.7 h
4	2039.8 h	351.1 h	29.7 h

Each TD3 case simulated with Shift was executed to $t = 6\,\mathrm{s}$ with 1×10^5 particle histories per cycle, 150 inactive cycles, and 250 active cycles. For the TML calculations, the CMFD mesh used a pincell resolution (51×51 mesh elements). All of the Shift calculations were executed on an ORNL compute cluster featuring dual-processor AMD Opteron 6378 CPUs with 16 cores per CPU, for a total of 32 cores per node, and 128 GB RAM per node. All Shift simulations were run on eight nodes (256 cores). The MPACT simulations were carried out for the same transient specifications up until $t = 10\,\mathrm{s}$. Additionally, the MPACT

benchmark only reported wall time using nine processors. The wall times reported by MPACT were then multiplied by a factor of nine to give an estimate for the number of core hours used. While these numbers are approximate and not directly comparable, we see that the Shift PCQM and TML implementations are roughly two and three orders of magnitude slower, respectively, than the MPACT implementation.

5. Conclusions and Future Work

The PCQM and TML methods implemented in Shift show benchmark results consistent with the same simulation executed with MPACT for all four TD3 cases when the stochastic noise after $t = 2\,\mathrm{s}$ is removed. The transient multilevel method applied to the TD3 problem smooths out stochastic noise both when compared visually and across common norm metrics. While a rigorous convervence analysis was not performed, the TML method allows much larger MC time steps to be taken and the PCQM, which decreases the number of expensive MC transport solves necessary to simulate a reactor transient. The TML reduced the overall simulation time by nearly an order of magnitude in the TD3 problems studied for this work. While these results are very promising, the robustness of the TML still needs to be tested with more rigours benchmarks that use 3D geometries, continuous-energy physics, feedback mechanisms and higher system dominance ratios.

The transient capabilities in Shift are still under development. Ongoing work includes efforts to improve the speed, accuracy, and robustness of these calculations. Specific improvements include implementing continuous-energy delayed neutron sampling, integrating feedback via Doppler broadening, and introducing novel techniques, such as parallel-in-time integration methods to the TML.

6. Acknowledgments

This work was supported by funding from the the Department of Energy Office of Nuclear Energy Spent Fuel and Waste Science and Technology program.

REFERENCES

[1] A. Zhu, Y. Xu, and T. Downar. "A Multlevel Quasi-Static Method for Pin-Resolved Transport Transient Reactor Analysis." *Nuclear Science and Engineering*, **volume 182** (2016).

[2] Z. M. Prince and J. C. Ragusa. "Multiphysics reactor-core simulations using the improved quasi-static method." *Annals of Nuclear Energy*, **volume 125**, pp. 186–200 (2019). URL https://www.sciencedirect.com/science/article/pii/S030645491830584X.

[3] Q. Shen, Y. Wang, D. Jabaay, B. Kochunas, and T. Downar. "Transient analysis of C5G7-TD benchmark with MPACT." *Annals of Nuclear Energy*, **volume 125**, pp. 107–120 (2018).

[4] B. L. Sjenitzer and J. E. Hoogenboom. "Dynamic Monte Carlo for Nuclear Reactor Kinetics Calculations." *Nuclear Science and Engineering*, **volume 175** (2013).

[5] M. W. Hackemack, J. C. Ragusa, D. P. Griesheimer, and J. M. Pounders. "A Monte Carlo Implementation of the Predictor-Corrector Quasi-Static Method." In *International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering (M&C 2013)*. American Nuclear Society (2013).

[6] Y. Jo, B. Cho, and N. Z. Cho. "Nuclear Reactor Transient Analysis by Continuous-Energy Monte Carlo Calculation Based on Predictor-Corrector Quasi-Static Method." *Nuclear Science and Engineering*, **volume 183** (2016).

[7] T. M. Pandya et al. "Implementation, Capabilities, and Benchmarking of Shift, a Massively Parallel Monte Carlo Radiation Transport Code." *Journal of Computational Physics*, **volume 308**, pp. 239–272 (2016).

[8] K. O. Ott and J. T. Madell. "Quasistatic Treatment of Spatial Phenomena in Reactor Dynamics." *Nuclear Science and Engineering*, **volume 26**(4) (1966).

[9] J. C. Gehin. *A Quasi-static Polynomial Nodal Method for Nuclear Reactor Analysis*. Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, MA (1992).

[10] S. Dulla, E. H. Mund, and P. Ravetto. "The quasi-static method revisited." *Progress in Nuclear Energy*, **volume 50** (2008).

[11] A. Zhu, Y. Xu, T. Saller, and T. Downar. "The Implementation and Analysis of the MOC and CMFD Adjoint Capabilities in the 2D-1D Code MPACT." In *Supercomputing in Nuclear Applications (SNA) and the Monte Carlo (MC) Method*, pp. 59–71. American Nuclear Society (2015).

[12] E. R. Wolters, E. W. Larsen, and W. R. Martin. "Hybrid Monte Carlo-CMFD Methods for Accelerating Fission Source Convergence." *Nuclear Science and Engineering*, **volume 174**(3), pp. 286–299 (2013). URL https://doi.org/10.13182/NSE12-72.

[13] K. Ott and R. Neuhold. *Introductory Nuclear Reactor Dynamics*. American Nuclear Society (1985). URL https://books.google.com/books?id=IAyMQgAACAAJ.

[14] J. Hou, K. N. Ivanov, V. F. Boyarinov, and P. A. Fomichenko. "OECD/NEA benchmark for time-dependent neutron transport calculations without spatial homogenization." *Nuclear Engineering and Design*, **volume 317**, pp. 177–189 (2017).

[15] T. M. Evans, A. Stafford, and K. T. Clarno. "Denovo–A New Three-Dimensional Parallel Discrete Ordinates Code in SCALE." *Nuclear Technology*, **volume 171**(8) (2010). URL https://www.osti.gov/biblio/984374.

[16] P. K. Romano, N. E. Horelik, B. R. Herman, A. G. Nelson, and B. Forget. "OpenMC: A state-of-the-art Monte Carlo code for research and development." *Ann of Nucl Energy*, **volume 82**, pp. 90–97 (2015).

[17] B. Kochunas, B. Collins, D. Jabaay, T. J. Downar, and W. R. Martin. "Overview of development and design of MPACT: Michigan parallel characteristics transport code." (2013). URL https://www.osti.gov/biblio/22212692.

[18] S. C. Shaner. *Development of High Fidelity Methods for 3D Monte Carlo Transient Analysis of Nuclear Reactors*. Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, MA (2018).

[19] M. A. Kreher, K. Smith, and B. Forget. "Direct Comparison of High-Order/Low-Order Transient Methods on the 2D-LRA Benchmark Problem." *Nuclear Science and Engineering*, **volume 0**(0), pp. 1–24 (2021). URL https://doi.org/10.1080/00295639.2021.1980363.

[20] E. S. Gonzalez and B. C. Kiedrowski. "C5G7-TD3 Transient Benchmark Results with Shift." In *Trans. Am. Nucl. Soc.*, volume 125, pp. 424–427 (2021).