PHYSOR-2006, ANS Topical Meeting on Reactor Physics
Organized and hosted by the Canadian Nuclear Society. Vancouver, BC, Canada. 2006 September 10-14

Accuracy of a Predictor-Corrector Quasi-Static Method for Space-Time Reactor Dynamics

Sandra Dulla[*1] , Ernest H. Mund[2] and Piero Ravetto[1]
[1] *Politecnico di Torino, Torino, Italy*
[2] *Université Libre de Bruxelles, Brussels, Belgium*

Abstract

The quasi-static method is a standard tool for the space-time solution of neutron transport problems in multiplying media. Its basic principle lies in a factorization of the angular flux into the product of two functions, 'amplitude' and 'shape', where the amplitude depends only on time (and contains the major part of the time-dependence) while the shape function depends on all variables, time included. The shape equation is solved on a long time scale, while the amplitude is determined on a short time scale. The factorization is made unique by proper normalization conditions for the shape function. Most implementations replace the basic equation (transport or diffusion) by the set of coupled amplitude and shape equations derived from the factorization, the so-called 'Improved Quasi-Static Method' (IQM). In this paper we describe an alternate approach already known for some time, called here 'Predictor-Corrector Quasi-Static Method' (PCQM), and we discuss its efficiency for both solid- and liquid fuel systems.

KEYWORDS: Reactor dynamics, quasi-static method

1. Introduction

The quasi-static (QS) method introduced by Henry at the end of the fifties has become a standard tool for the solution of space-time neutron transport problems [1–3]. Most of the time it is used in the algorithmic form developed by Ott and Meneley called Improved Quasi-Static Method (IQM) [4]. Its basic principle lies in a factorization of the time-dependent neutron flux into the product of two functions called 'amplitude' and 'shape':

$$\phi(\mathbf{r}, E, \mathbf{\Omega}, t) \;=\; T(t)\, \psi(\mathbf{r}, E, \mathbf{\Omega}, t). \tag{1}$$

The amplitude function $T(t)$ depends only on time and provides the bulk information about the power change, while the shape function $\psi(\mathbf{r}, E, \mathbf{\Omega}, t)$, depending on all variables describes the (time-dependent) power profile deformation. Obviously, factorization (1) is not unique. To be made unique, some numerical constraint on the shape must be kept during the transient. Equations for the amplitude and for the shape are derived from the full model. The amplitude

[*]Corresponding author, Tel. +39-011-564-4443, Fax. +39-011-564-4499, E-mail: sandra.dulla@polito.it

equations are nothing else that the familiar point kinetics (PK) equations, usually solved on a small time-step h (the micro time-step). The shape equation is the basic time-dependent neutron transport equation with coupling terms involving the amplitude. Time-stepping in the numerical solution of the shape equation is larger. We call this the macro time-step $\Delta T = nh$, where n is an integer.

The rationale of the QS method is to spare computer time in relying more on the information provided by a small number of ODEs (the PK equations) than on the information contained in the basic PDEs. However, since the coefficients of the PK equations (reactivity, prompt neutron lifetime, effective delayed-neutron fractions, etc. . .) closely depend on the flux distribution, the accuracy of the latter is crucial. In IQM the PK parameters are evaluated using the solution to the shape equation on ΔT. Hence, besides the normalization constraint an elaborate iterative scheme is used to solve the (nonlinear) coupled set of point-kinetics and shape equations. Despite the intricacies of this nonlinear problem, there is a hope that computer savings may be achieved. Recent numerical experiments show that physical situations exist where the convergence of the iterative method is so slow (or problematic) that the overall computer time for some reactor transients is larger than the computer time entailed by a direct solution technique using the most suitable implicit numerical algorithm [3].

Another approach to the QS method exists that also takes advantage of the 'amplitude/shape' splitting. Although known for some time (see [5]), it has received little attention with regard to computation efficiency. Instead of working with the amplitude and shape, this approach keeps the angular flux as basic unknown and uses the shape as a convenient tool to define the amplitude. As a consequence the computation of the angular flux and reaction rates results from a predictor-corrector algorithm (hence the name PCQM) that can easily be made adaptive for error control. This paper describes the PCQM algorithm and compares its efficiency with that of IQM, in particular for problems sensitive with regard to flux tilt. Section 2 recalls the IQM method in the transport and continuous energy framework. Adaptation to diffusion theory and to multigroup equations is straightforward. Section 3 outlines the alternate PCQM method. Section 4 discusses numerical results.

2. The Improved Quasi-static Method

Consider the time-dependent neutron transport and delayed-neutron precursors equations, for a stationary fuel in some physical domain $V \in I\!R^3$:

$$\frac{1}{v}\frac{\partial \phi}{\partial t} = \mathcal{L}(t)\,\phi(t) + \frac{1}{4\pi}\sum_i \chi_i(E)\lambda_i C_i(\mathbf{r},t) + S(\mathbf{r},E,\mathbf{\Omega},t), \tag{2}$$

$$\frac{\chi_i}{4\pi}\frac{\partial C_i}{\partial t} = -\frac{\chi_i}{4\pi}\lambda_i C_i(\mathbf{r},t) + J_i^d(t)\phi(t), \qquad i = 1,\ldots,I, \tag{3}$$

where $\phi(t)$ $(\equiv \phi(\mathbf{r},E,\mathbf{\Omega},t))$ and $C_i(\mathbf{r},t)$ denote the angular neutron flux and the delayed neutron precursors concentrations. The symbol $\mathbf{\Omega}$ refers to any direction in the physical space, and the particles kinetic energy range is $(0, E_{max})$. Equation (2) contains an external neutron source S. Although this term is usually absent, it might represent a spallation neutron source of an accelerator driven system (ADS). The symbol $\mathcal{L}(t)$ denotes the time-dependent transport operator including streaming, absorption, scattering and prompt fission:

$$\mathcal{L}(t)\,\phi(t) = \mathcal{J}_p(t)\,\phi(t) - \mathcal{K}(t)\,\phi(t), \tag{4}$$

with

$$\mathcal{K}(t)\,\phi(t) = \mathbf{\Omega}\cdot\nabla\phi(t) + \Sigma_t(t)\phi(t) - \iint dE'd\Omega'\Sigma_s(\mathbf{r},E',\mathbf{\Omega}'\rightarrow E,\mathbf{\Omega},t)\phi(\mathbf{r},E',\mathbf{\Omega}',t). \quad (5)$$

Prompt and delayed neutron production rates $\mathcal{J}_p(t)\phi(t)$ and $\mathcal{J}_i^d(t)\phi(t)$ are given by:

$$\begin{aligned}\mathcal{J}_p(t)\phi(t) &= (1-\beta)\frac{\chi_p(E)}{4\pi}\iint dE'd\Omega'\nu\Sigma_f(\mathbf{r},E',t)\phi(\mathbf{r},E',\mathbf{\Omega}',t),\\ \mathcal{J}_i^d(t)\phi(t) &= \beta_i\frac{\chi_i(E)}{4\pi}\iint dE'd\Omega'\nu\Sigma_f(\mathbf{r},E',t)\phi(\mathbf{r},E',\mathbf{\Omega}',t), \qquad i=1\ldots I, \quad (6)\end{aligned}$$

I being the number of delayed neutron precursor families. We consider vacuum boundary conditions and, for the sake of simplicity, we assume there is no external neutron source and the physical system is in steady-state, criticality conditions being met. The initial neutron flux ϕ_0 $(\equiv\phi(\mathbf{r},E,\mathbf{\Omega},0))$ therefore satisfies the transport equation:

$$\mathcal{L}_0\phi_0 + \sum_i \mathcal{J}_{i0}^d\phi_0 = 0, \quad (7)$$

where $\mathcal{L}_0$ and $\mathcal{J}_{i0}^d$ denote the steady-state transport and delayed neutrons operators.

We also introduce the steady-state neutron importance $\psi_0^\star(\mathbf{r},E,\mathbf{\Omega})$ solution to:

$$\mathcal{L}_0^\star\psi_0^\star + \sum_i \mathcal{J}_{i0}^d\psi_0^\star = 0, \quad (8)$$

where $\mathcal{L}_0^\star$ denotes the adjoint of $\mathcal{L}_0$, with adequate boundary conditions.

At $t = 0$ a transient is inserted into the system. The system then behaves according to Eqs. (2)-(3) with initial condition $(\phi_0(\mathbf{r},E,\mathbf{\Omega}), C_i(\mathbf{r},0))$. We might of course use any suitable time-integration method for these equations. Instead, we factorize the neutron flux into the product (1). Obviously such splitting is perfectly arbitrary, since one could multiply T by some factor and divide ψ by the same factor. To make the splitting unique we impose a constraint on the shape function that very naturally writes:

$$<\psi_0^\star,\frac{1}{v}\psi> := \iiint d\mathbf{r}\,dE\,d\Omega\;\psi_0^\star(\mathbf{r},E,\mathbf{\Omega})\,\frac{1}{v}\,\psi(\mathbf{r},E,\mathbf{\Omega},t) \;=\; c\equiv<\psi_0^\star,\frac{1}{v}\phi_0>\,. \quad (9)$$

Taking the partial derivative of (1) with respect to time gives:

$$\frac{\partial\phi}{\partial t} \;=\; \frac{dT}{dt}\,\psi(\mathbf{r},E,\mathbf{\Omega},t) \;+\; T(t)\frac{\partial\psi}{\partial t}. \quad (10)$$

Introducing (1) and (10) into (2)-(3) without external neutron source, and rearranging terms, one gets:

$$\mathcal{L}(t)\psi(\mathbf{r},E,\mathbf{\Omega},t) \;+\; \frac{1}{4\pi}\sum_i \lambda_i\chi_i(E)C_i(\mathbf{r},t)\,\frac{1}{T(t)} = \frac{1}{v}\frac{\partial\psi}{\partial t} \;+\; \frac{1}{v}\,\psi(\mathbf{r},E,\mathbf{\Omega},t)\,\frac{1}{T(t)}\,\frac{dT}{dt}. \quad (11)$$

Equation (11) is the shape equation. It delivers $\psi(\mathbf{r},E,\mathbf{\Omega},t)$ provided $T(t)$ is known. Notice that the delayed neutron precursors concentrations $C_i(\mathbf{r},t)$ themselves depend on the neutron flux $\phi(t)$ through Eq. (3).

Subtracting the steady-state adjoint equation (8) multiplied by ψ from (11) multiplied by $\psi_0^\star$, and integrating over all variables, one gets the system of $(I+1)$ ODEs for the amplitude and the concentrations of the neutron precursors :

$$\left\{ \begin{array}{l} \dfrac{dT}{dt} = \dfrac{\rho(t) - \overline{\beta}(t)}{\Lambda(t)} T(t) + \displaystyle\sum_i \lambda_i c_i(t) \\ \dfrac{dc_i}{dt} = -\lambda_i c_i(t) + \dfrac{\overline{\beta}_i(t)}{\Lambda(t)} T(t), \qquad i = 1, \ldots, I, \end{array} \right. \tag{12}$$

with the set of parameters

$$F(t) = \iiint d\mathbf{r}\, dE\, d\Omega\, \psi_0^\star(\mathbf{r}, E, \mathbf{\Omega})\, \chi_p(E) \iint dE'\, d\Omega'\, \nu\Sigma_f \psi(\mathbf{r}, E', \mathbf{\Omega}', t),$$

$$\Lambda(t) = \frac{1}{F(t)} \iiint d\mathbf{r}\, dE\, d\Omega\, \psi_0^\star(\mathbf{r}, E, \mathbf{\Omega})\, \frac{1}{v}\, \psi(\mathbf{r}, E, \mathbf{\Omega}, t), \tag{13}$$

$$c_i(t) = \frac{1}{F(t)} \frac{1}{\Lambda(t)} \iiint d\mathbf{r}\, dE\, d\Omega\, \psi_0^\star(\mathbf{r}, E, \mathbf{\Omega})\, \chi_i(E) C_i(\mathbf{r}, t), \tag{14}$$

$$\rho(t) = \frac{1}{F(t)} \iiint d\mathbf{r}\, dE\, d\Omega\, \psi_0^\star(\mathbf{r}, E, \mathbf{\Omega})\, \Delta\mathcal{L}^\star(t) \psi(\mathbf{r}, E, \mathbf{\Omega}, t), \tag{15}$$

$$\overline{\beta}_i(t) = \frac{1}{F(t)}\, \beta_i \iiint d\mathbf{r}\, dE\, d\Omega\, \psi_0^\star(\mathbf{r}, E, \mathbf{\Omega})\, \Delta\mathcal{J}(t) \psi(\mathbf{r}, E, \mathbf{\Omega}, t), \tag{16}$$

and

$$\overline{\beta}(t) = \sum_{i=1}^{I} \overline{\beta}_i(t). \tag{17}$$

Equations (12)-(17) are the PK equations with its integral parameters: reactivity $\rho(t)$, prompt neutron lifetime $\Lambda(t)$ and effective delayed neutron fractions $\overline{\beta}_i(t)$. The shape and amplitude equations form a set of non-linear equations. The solution of Eq. (11) implies the knowledge of $T(t)$ while the parameters in Eq. (12) depend on $\psi(\mathbf{r}, E, \mathbf{\Omega}, t)$. The most straightforward solution to (11)-(12) therefore is an iterative procedure with macro time-step ΔT for the evaluation of the shape ψ, and micro time step h for the evaluation of the amplitude $T(t)$.

In the first algorithmic step one evaluates the PK parameters (13)-(17) at time ΔT, assuming the shape function has *not* been modified from its steady-state value ϕ_0. Once the PK parameters are known, the amplitude equations (12) may be solved approximately using a stiff ODE system solver with time-step h, up to $t = \Delta T$. Knowing the amplitude at $t = \Delta T$ allows the integration of the shape equation and a new estimation of the point kinetics parameters at the first macro time-step. The nonlinear system (11)-(12) is solved iteratively on the first macro time-step up to some given convergence criterion. The constraint (9) must be satisfied at any time.

In substance, this is the IQM algorithm of Ott and Meneley where the partial derivative of the shape is approximated through (see [4]):

$$\frac{\partial \psi(\mathbf{r}, E, \mathbf{\Omega}, t)}{\partial t} \approx \frac{\psi(\mathbf{r}, E, \mathbf{\Omega}, t) - \psi(\mathbf{r}, E, \mathbf{\Omega}, t - \Delta T)}{\Delta T}, \tag{18}$$

corresponding to an *implicit* Euler algorithm with time-step ΔT and error in the time integration $O(\Delta T^2)$. Provided the transient does not induce severe flux tilts, the time integration error on the shape will be small, the global error being dominated by the error induced by the transport equation solver.

3. The Predictor-Corrector Quasi-static Method

We keep the neutron flux splitting (1) and constraint (9) but, instead of working with shape ψ and amplitude T, we use the basic transport equations (2)-(3) and the PK equations (12). In other terms, the shape here is a convenient tool to extract an 'amplitude' from the solution of the basic equations but *not* an unknown quantity as in IQM.

Again, we consider two time-scales: ΔT for the angular flux ϕ, and h for the amplitude T. Ideally, the ratio $\Delta T/h$ should be as large as possible. For a prompt critical transient in a loosely coupled reactor, flux tilts might impose low values of this ratio.

On the macro time-scale ΔT, we solve Eqs. (2)-(3) using the implicit Euler approximation:

$$\begin{aligned} \frac{\partial \phi(\mathbf{r},E,\mathbf{\Omega},t)}{\partial t} &\approx \frac{\phi(\mathbf{r},E,\mathbf{\Omega},t)-\phi(\mathbf{r},E,\mathbf{\Omega},t-\Delta T)}{\Delta T}, \\ \frac{\partial C_i(\mathbf{r},t)}{\partial t} &\approx \frac{C_i(\mathbf{r},t)-C_i(\mathbf{r},t-\Delta T)}{\Delta T}, \end{aligned} \tag{19}$$

On the micro time-scale h we solve the amplitude equations (12) as indicated before. The PCQM algorithm runs as follows:

- **Step 1.** Evaluate the steady-state direct and adjoint angular fluxes ϕ_0 and $\psi_0^\star$, solution to Eqs. (7)-(8) and compute the constant c such that $c = <\psi_0^\star, \frac{1}{v}\phi_0>$.

- **Step 2.** Using the implicit Euler scheme (19), evaluate the angular flux and delayed precursors concentration at time $t = \Delta T$ denoted respectively $\widetilde{\phi}(\mathbf{r},E,\mathbf{\Omega},\Delta T)$ and $\widetilde{C}_i(\mathbf{r},\Delta T)$. These are *predicted* values with (presumably) some error on the amplitude because of the (large) time-step ΔT. This error, however will be corrected at the next step.

  One now evaluates the quantity $z = \frac{1}{c} <\psi_0^\star, \frac{1}{v}\widetilde{\phi}>$. One verifies easily that

  $$\psi(\mathbf{r},E,\mathbf{\Omega},\Delta T) := \frac{1}{z}\widetilde{\phi}(\mathbf{r},E,\mathbf{\Omega},\Delta T),$$

  satisfies the constraint (9) and, hence is an approximate shape function.

- **Step 3.** Next, one evaluates the PK parameters (13)-(17) at $t = \Delta T$ using the shape function ψ computed at Step 2, and solves the amplitude equations on the micro time-scale, using a standard method for stiff equations with the initial conditions $T(0) = 1$, and $c_i(0) = \frac{1}{\lambda_i}\frac{\overline{\beta}_i(0)}{\Lambda(0)}$, $i = 1,\ldots,I$. This computation gives approximate values for $T(\Delta T)$ and $c_i(\Delta T)$ after $\Delta T/h$ steps with errors $O(h^p)$ depending on the ODE solver.

- **Step 4.** Once the amplitude at time $t = \Delta T$ is known, one gets *corrected* values of the angular flux and of the precursors concentrations through:

  $$\phi(\mathbf{r},E,\mathbf{\Omega},\Delta T) := \psi(\mathbf{r},E,\mathbf{\Omega},\Delta T)\, T(\Delta T), \tag{20}$$

  and

  $$C_i(\mathbf{r},\Delta T) = \frac{1}{1+\lambda_i\Delta T}C_i(\mathbf{r},0) + \frac{\beta_i\Delta T}{1+\lambda_i\Delta T}\iint dE'd\Omega'\nu\Sigma_f\phi(\mathbf{r},E',\mathbf{\Omega}',\Delta T), \tag{21}$$

  where (21) results from the implicit Euler approximation applied to Eq. (3). Step 4 completes the determination of an approximate solution of the angular flux and precursor concentrations at time ΔT. The same procedure must be applied successively (starting from step 2) to all subsequent macro time-steps, up to the end of the transient.

Unlike IQM this algorithm is *linear*. It means that once the angular flux and delayed neutron precursors concentrations have been 'corrected', there is no way to further improve the approximation through some iterative process. The only improvement would be by halving the macro time-step, applying twice the implicit Euler scheme with time-step $\Delta T/2$, evaluating refined PK parameters and extrapolating the 'corrected' values of angular flux with time-steps ΔT and $\Delta T/2$. The splitting into amplitude and shape here is a convenient trick to put the stiffness of the problem on a reduced set of equations, the amplitude equations, the basic transport equation being used solely to catch the space distortion of the flux and evaluate the PK parameters.

4. Numerical Results

To compare IQM and PCQM we have considered a critical reflected reactor system in cylindrical (r, z) geometry, in $3-$group *diffusion* approximation. A 'step' perturbation of the removal cross sections in groups one and two in a localized core region introduces a -2411 pcm reactivity. This perturbation is compensated by a homogeneous perturbation of the multiplication properties of the medium to retain criticality. Table 1 summarizes the results obtained in terms of computation times elapsed during the shape and flux calculation for this transient. The evaluation of the computation effort has been focused on the update procedure since PK has roughly the same impact on PCQM and IQM execution times. Tolerances for the flux and the shape calculations have been assumed equal. In the case of one update the values provided by IQM are basically PK results. A shape update has been performed at the end of the time interval to compare the flux evaluations performed in the two algorithms.

Table 1: Total computational times for IQM and PCQM algorithms (normalized to the most time consuming). N is the number of updates.

N	ΔT [ms]	PCQM	IQM	N	ΔT [ms]	PCQM	IQM
1	1.0	0.306	1.000	25	0.04	1.528	3.613
2	0.5	0.444	0.985	100	0.01	2.946	6.727
4	0.25	0.598	1.544	500	0.002	9.376	8.670
10	0.1	0.966	2.345	1000	0.001	16.655	15.516

The PCQM method is more efficient for small numbers of shape updates. This advantage progressively disappears with larger numbers of flux updates. For this particular transient, the explanation lies in the fact that once the shape is established, IQM is faster because it requires just one solution of the full spatial model equation. We stress again that the rationale of the QS method lies in a small number of shape recalculations. In this situation, PCQM avoids the iteration to fulfill the normalization constraint and thus is faster in updating the shape functions.

Our comparison between the two methods has also been focused on the iterations characterizing the shape and flux calculation. In this specific case of a compensated transient, in which large spatial phenomena are involved with very little effect on the total power (i.e. the "exact" amplitude function is always very close to its initial value), the number of outer iterations required for the solution of the flux problem for PCQM and the maximum number of outer iterations required for the shape problem for IQM is almost the same. As a consequence, the inclusion of the normalization iteration for the IQM turns out in a slowing down of the procedure. These considerations are proved by observing the case of 10 shape updates, presented in Table 2.

Table 2: Comparison of the number of iterations for IQM and PCQM algorithms in the case of 10 shape updates. OI: number of outer iterations on the energy groups for PCQM; NI: number of iterations to fulfill the normalization condition for IQM; MOI: maximum number of outer iterations observed in the shape calculation for IQM (usually occurring during the first normalization iteration).

shape update		1	2	3	4	5	6	7	8	9	10
IQM	NI	10	8	8	8	7	5	5	4	4	3
	MOI	7	5	4	3	3	2	2	2	2	2
PCQM	OI	7	5	4	4	3	3	2	2	2	2

Figure 1 displays the evolution of power obtained with full space-time integration of the problem (solid line). This is compared with the results obtained with the IQM and PCQM methods using the same time interval ΔT for the shape update as the one used for the full problem inversion. Deviation of QS results (both IQM and PCQM) from the *reference* is basically due to the fact that in the *reference* case the space-time problem is solved over ΔT, while:

- in the PCQM algorithm the space-time problem is followed by a calculation of the amplitude evolution which is based on the use of kinetic parameters, that can be viewed as an *approximation* of the exact values through perturbation theory;
- in the IQM algorithm firstly the amplitude is evaluated using kinetic parameters associated to the shape function at the beginning of the time step, then a different space-time problem is solved, iterated on the value of the derivative of the amplitude function itself.

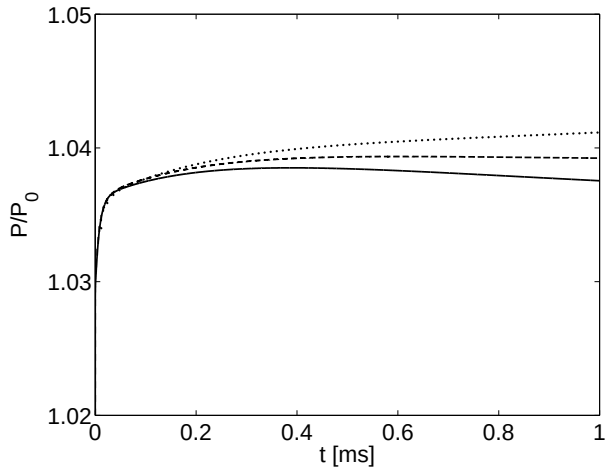
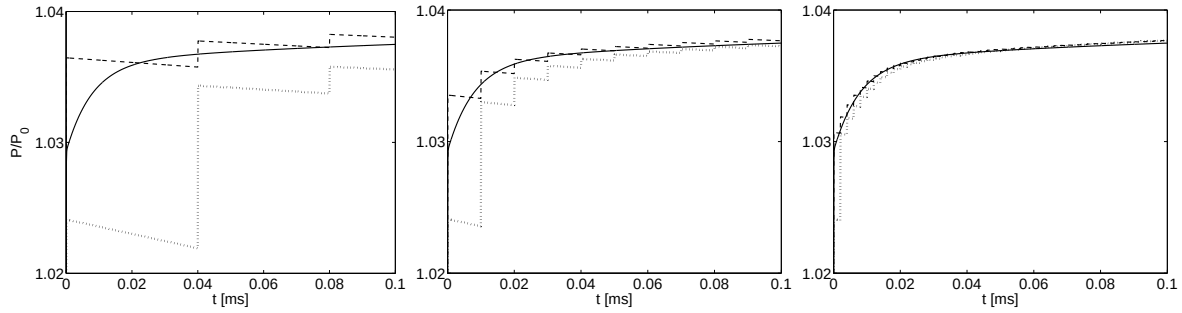


Figure 1: P/P_0 with reference solution (solid line), IQM (dotted) and PCQM (dashed).

It is clear that the two methods solve different numerically-discretized equations, and this might cause discrepancies in the results, while all the methods converge to the exact solution when ΔT goes to zero.

Furthermore, PCQM uses a shape function and corresponding kinetic parameters on ΔT that provide better information on the transient, while the shape used to compute the kinetic parameters for the IQM amplitude calculation retains no information on the evolution of the transient for the interval over which it is used. This peculiarity can be clearly observed in Fig. 2, where the accuracy of the results provided by the IQM and PCQM methods is compared to the reference result, focusing on some values of ΔT. One should notice the difference between the reference result and the two QS approaches. In particular, since PCQM generates kinetic parameters using a shape that contains more 'information' on the evolution of the transient, the corresponding amplitude is closer to the exact solution. The IQM method in comparison delivers the kinetic parameters with a certain delay, using a shape function obtained from a space-time calculation at the beginning of the time interval.



(a) $\Delta T = 0.04$ ms. (b) $\Delta T = 0.01$ ms. (c) $\Delta T = 0.002$ ms.

Figure 2: Comparison of IQM (dotted line) and PCQM (dashed line) with reference solution (solid line) for different values of ΔT.

A completely different situation arises when the transient involves large modification of the total power, and consequently of the amplitude function. In this case the solution of the full space-time model requires a large number of outer iterations, since the flux along the time interval ΔT experiences both shape and amplitude effects. The shape model solved in the IQM, on the other side, is affected only by shape effects, since the amplitude effects are taken into account by the amplitude function itself and by the additional iteration required for the fulfillment of the normalization condition. As a consequence, the performances change dramatically and the calculation effort turns out to be larger for the PCQM. This effect is very large when a small number of recalculations are involved, and thus large changes in power, while it decreases for increasing number of updates. This situation is clearly observable when considering the non-compensated transient associated only to the localized increase in absorption analyzed previously. The reactor power decreases almost to half its initial values in 1 second; Table 3 reports the data concerning the computational effort for the case of 2 and 10 shape updates. It clearly turns out that a too small number of shape updates results in a disappointing performance on PCQM with respect to IQM.

In order to avoid situations in which PCQM is slower than IQM, two different approaches can be envisaged. The first option could be to perform transient calculations with a minimum number of updates that can justify the use of PCQM instead on IQM, even if the optimum is difficult to be predetermine. An alternative is speeding up the PCQM flux convergence by rescaling the initial flux guess by an approximation of the new power level, and thus partially

Table 3: Comparison of the number of iterations and computational time for IQM and PCQM algorithms in the case of 2 and 10 shape updates. Times are renormalized to the fastest calculation.

updates	IQM			PCQM		mod-PCQM	
$N = 2$	NI	MOI	time	OI	time	OI	time
1	8	14	15.494	30	17.906	8	4.065
2	8	5	12.959	27	16.023	11	6.006
total	–	–	28.453	–	33.929	–	10.171
$N = 10$	NI	MOI	time	OI	time	OI	time
2	10	7	7.894	12	6.188	7	3.359
4	10	4	6.776	11	5.659	4	1.647
6	10	3	6.006	11	5.594	3	1.235
8	10	2	5.894	11	5.594	2	1.006
10	10	2	5.947	11	5.535	2	1.000
total	–	–	68.040	–	59.434		20.264

compensating the amplitude effects that the flux is experiencing. This can be done as follows:

- solve the PK equations on ΔT with the parameters associated to the previous shape;
- use the evaluated amplitude function to rescale the flux guess;
- solve the flux problem over ΔT with the improved initial guess and normalize it in order to obtain a proper shape function;
- compute the point kinetic parameters with the updated shape and perform the calculation over ΔT with the PCQM algorithm.

The computation effort associated to PK evaluations is doubled but the increased cost is largely compensated by the speeding up of the flux solution. The transient is now re-evaluated with the new procedure, referred to as mod-PCQM, and the corresponding results are reported in Table 3, showing that the computation time is greatly reduced.

The PCQM algorithm has been applied to the kinetics of fluid fuel systems. The PK model for such reactor systems needs to be properly adapted [3] and the inclusion of QS requires the fulfillment of a normalization condition on the shape of the flux and of all precursor families. Consequently, iterations must be performed to solve the non-linearity of the model implying complications in the algorithm with obvious effects on computation times. It is then interesting to test whether PCQM can be of help to reduce the computation effort. The algorithm has been implemented in a multigroup diffusion cylindrical (r, z) code developed for the dynamics of molten-salt reactors and it is compared to the IQM scheme.

The transient considered for the analysis is induced by a combined modification of cross sections in a small fraction of the reactor, associated to a -863 pcm reactivity change. Since the perturbation has an impact on fission production, both the flux and delayed neutron precursors shapes are affected. The computation times and iteration numbers associated to IQM and PCQM are reported in Table 4. One observes that the promising performance of PCQM (for the solid fuel system) is not reproduced in the fluid fuel case. In absence of speeding PCQM

is much slower than IQM, and the introduction of a proper initial guess for flux and precursors decreases significantly the computation times, even if the performance of mod-PCQM still remains unsatisfactory with respect to IQM when few shape calculations are considered.

The observation of the power prediction allows some further considerations. With increasing N the results obtained with IQM are largely modified, while PCQM with $N = 1$ provides a results which is very close to the most accurate, obtained with 25 recalculations. As a consequence, the comparison of computational times can be consistently made on the basis of the same quality of results, thus comparing the IQM with 25 updates and PCQM with 1 shape recalculation. This comparison again shows the improved performance obtained with PCQM, mainly due to the possibility to provide more suitable kinetic parameters for the time dependent simulation, as discussed before.

Table 4: Comparison of the number of outer iterations and computational time for IQM, PCQM and mod-PCQM for a transient in a fluid-fuel system. Times are renormalized as in Table 3. In the case of 10 and 25 updates, only total data are given.

updates	IQM			PCQM		mod-PCQM	
$N = 1$	NI	MOI	time	OI	time	OI	time
1	8	11	0.109	278	0.984	78	0.258
$N = 2$	NI	MOI	time	OI	time	OI	time
1	8	11	0.108	247	0.876	69	0.226
2	8	2	0.053	277	1.000	3	0.012
total	–	–	0.161	–	1.876	–	0.238
$N = 10$	NI	MOI	time	OI	time	OI	time
total	3 (first)	9 (first)	0.278	240 (last)	8.074	95 (first)	0.395
$N = 25$	NI	MOI	time	OI	time	OI	time
total	4 (first)	9 (first)	0.627	203 (last)	17.104	86 (first)	0.436

References

1) A.F. Henry, "The Application of Reactor Kinetics to the Analysis of Experiments," Nucl. Sci. Engrg., **3**, 52–70 (1958).

2) S. Goluoglu, H.L. Dodds, "A Time-Dependent, Three-Dimensional Neutron Transport Methodology," Nucl. Sci. Engrg., **139**, 248–261,(2001).

3) S. Dulla, "Models and Methods in the Neutronics of Fluid Fuel Reactors," PhD thesis, Politecnico di Torino (2005).

4) K.O. Ott, D.A. Meneley, "Accuracy of the Quasistatic Treatment of Spatial Reactor Kinetics," Nucl. Sci. Engrg., **36**, 402–411 (1969).

5) J.C. Gehin, "A Quasi-Static Polynomial Nodal Method for Nuclear Reactor Analysis," PhD thesis, Massachusetts Institute of Technology (1992).