Matrix Equations and Tensor Techniques
IX
Perugia, September 9-10, 2021

A μ-mode integrator for solving evolution equations in Kronecker form

Marco Caliari *Fabio Cassini* Lukas Einkemmer

Alexander Ostermann Franco Zivcovich

Department of Mathematics, University of Trento,
Via Sommarive 14 – Povo (TN), Italy

Due to the importance of simulation in various fields of science and engineering, devising efficient numerical methods for solving high–dimensional evolutionary partial differential equations is of considerable interest. In this talk, we present a μ-mode integrator for computing the solution of stiff evolution equations. It is based on a d-dimensional splitting approach and it suitably combines, in a tensor framework, one-dimensional matrix exponentials (usually precomputed).

We show that our integrator solves exactly linear problems in Kronecker form with time-invariant coefficients, i.e. problems which can be written as

$$\mathbf{u}'(t) = M\mathbf{u}(t), \quad \mathbf{u}(0) = \mathbf{u}_0 \tag{1}$$

where

$$M = \sum_{\mu=1}^{d} A_{\otimes\mu}$$

and

$$A_{\otimes\mu} = I_d \otimes \cdots \otimes I_{\mu+1} \otimes A_\mu \otimes I_{\mu-1} \otimes \cdots \otimes I_1,$$

being A_μ an $n_\mu \times n_\mu$ matrix and I_μ the identity matrix of size n_μ. More in detail, it computes efficiently the exact solution of (1), i.e. $\mathbf{u}(t) = \exp(tM)\mathbf{u}_0$, by means of tensor techniques without explicitly forming the matrix M. This scheme can also be used as a building block for numerically solving more general classes of PDEs compared to (1), for example in the context of a splitting method or an exponential integrator.

We further explain how the needed actions of matrix exponentials can be implemented efficiently on modern computer systems, such as multithreaded CPUs and GPUs. In particular, the overall computational cost of our algorithm is $\mathcal{O}(N \max_\mu n_\mu)$, with $N = n_1 \cdots n_d$, while the storage requirement scales as $\mathcal{O}(N)$. Hence, the scheme is ideally suited to modern hardware, especially for GPUs.

We finally illustrate the features and the performances of the μ-mode integrator, both on CPUs and on GPUs, by numerically solving a range of problems from physics, such as three-dimensional heat equations (see Figure 1) and three-dimensional linear and nonlinear Schrödinger equations. In particular, we show that our integrator can significantly outperform numerical methods well established in the field and that we can obtain performance improvements between a factor of 10 and 20 by performing computations on GPUs rather than on CPUs.

If time allows, we also present how μ-mode products can be employed to compute spectral transforms efficiently even if no *fast* transform is available. This technique is useful, for example, in the context of a Hermite pseudospectral method.

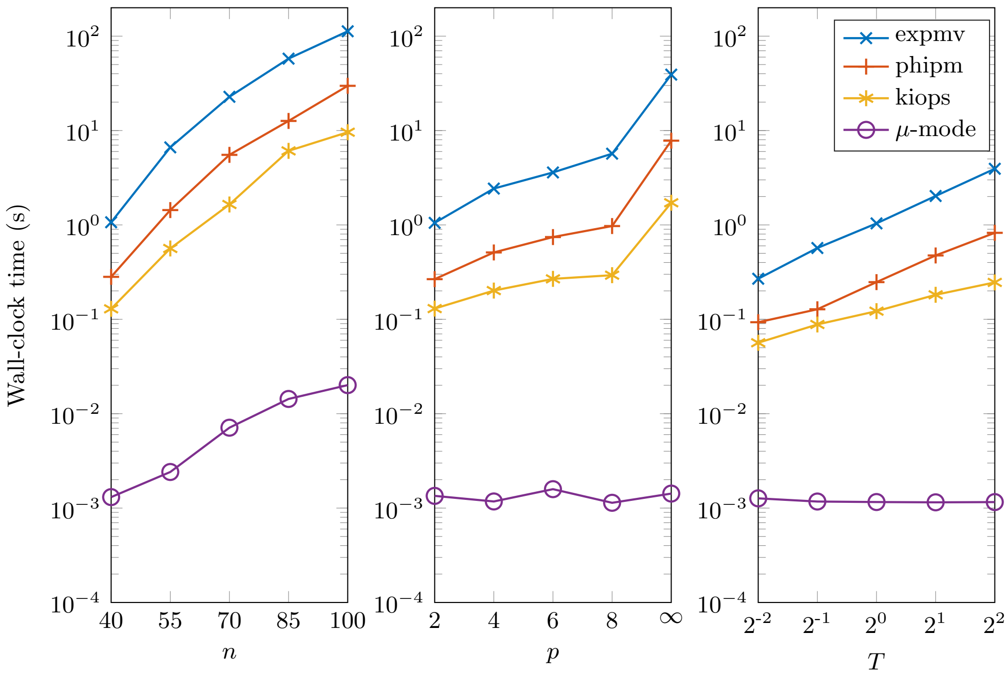

Figure 1: The wall-clock time for solving a three–dimensional heat equation is shown as a function of the size $n_\mu = n$ (left), of the order of the finite difference scheme p (middle), and of the final time T (right).

References

[1] M. Caliari, F. Cassini, L. Einkemmer, A. Ostermann, F. Zivcovich, *A μ-mode integrator for solving evolution equations in Kronecker form*, Submitted.