YIC2013
Second ECCOMAS Young Investigators Conference
2–6 September 2013, Bordeaux, France

Development of a parallel third order explicit multirate scheme

B. Seny [a,*], J. Lambrechts [a], V. Legat [a], J.-F. Remacle [a]

[a] Institute of Mechanics, Materials and Civil Engineering (iMMC), Université catholique de Louvain (UCL)
Bâtiment Euler, Avenue Georges Lemaître 4, 1348 Louvain-la-Neuve, Belgium

*bruno.seny@uclouvain.be

Abstract. *Multirate schemes aims at circumventing the global stability restriction of classical explicit methods. The idea is to gather mesh elements in groups that satisfy local stability conditions. The transition between groups has to be accommodated to preserve convergence and conservation properties. The extension of these strategies to the parallel framework is challenging since the computational load varies at each stage of the algorithm. Here we focus on the parallel implementation of a third order multirate strategy for discontinuous Galerkin simulations.*

Keywords: Multirate time stepping; parallelization; high performance computing; partitioning strategies.

1 INTRODUCTION

Multirate explicit schemes have the vocation of reducing the important limitations due to the severe restrictions on the time step imposed by the Courant-Friedrichs-Lewy (CFL) condition. Important gaps between the smallest and largest element-wise stable time step may occur for problems with unstructured meshes and/or highly varying advection speeds. The idea is to organize the mesh elements in groups that share a local stability limit and ensure a adequate transition between them by inserting buffer layers. Conservative multirate schemes of second-order have been proposed by Constantinescu [3] and a third order scheme has been developed by Schlegel [1]. These methods have proven to be very attractive in the discontinuous Galerkin (DG) framework [4]. Extending these multirate strategies to the parallel context is challenging since a classical mesh partitioning is not adequate anymore. Indeed, the computational and communication costs are not constant at each stage of the algorithm. Therefore we use a multi-constraint heuristic to find the best compromise between load-balancing constraints and minimization of communication overheads. The purpose of the present work is to extend the parallel strategy developed in [5] for the second-order multirate schemes of Constantinescu to the third order multirate method of Schlegel.

2 A THIRD ORDER MULTIRATE EXPLICIT RUNGE-KUTTA SCHEME

The multirate strategy proposed by Schlegel et al. [1] borrows some ideas of an implicit-explicit (IMEX) time integration scheme [2]. The idea is to use an explicit method for both the slow and fast components. Based on a classical explicit Runge-Kutta (ERK) method they construct an inner and outer method that use an extra number of stages and that are used together as a partitioned Runge-Kutta (PRK) method. The elements treated by the inner method have a stable time step twice smaller than the elements treated by the outer method. Both methods have the same number of stages but they don't require the same amount of substantial operations (evaluation of the steady-state residual). Accordingly, considerable speedups compared to the singlerate method can be obtained.

2.1 Multirate groups for the discontinuous Galerkin framework

Due to the compact nature of DG elements, one element only needs the information of it's direct neighbors to compute the interface fluxes. The mesh elements may thus be organized in multirate groups that share the same features. Consider that the smallest stable time step associated to a mesh is Δt_m. Elements that have a stable time step belonging to a range defined as $[2^n \Delta t_m, \cdots, 2^{n+1} \Delta t_m[$ are allocated to a same group. To accommodate the

Table 1: Butcher tableaus of the outer buffer method (a) and the inner buffer method (b) for RK43S.

(a)

0										
1/4	1/4									
1/4	1/4	0								
1/2	1/2	0	0							
1/2	**1/2**	0	0	0						
1/2	**-1/6**	0	0	0	**2/3**					
3/4	1/12	0	0	0	1/6	1/2				
3/4	1/12	0	0	0	1/6	1/2	0			
1	1/3	0	0	0	-1/3	1	0	0		
1	**1/3**	0	0	0	**-1/3**	**1**	0	0	0	
	1/6	0	0	0	**1/3**	**1/3**	0	0	0	**1/6**

(b)

0										
1/4	**1/4**									
1/4	**-1/12**	**1/3**								
1/2	**1/6**	**-1/6**	**1/2**							
1/2	1/12	1/6	1/6	1/12						
1/2	**1/12**	**1/6**	**1/6**	**1/12**	0					
3/4	**1/12**	**1/6**	**1/6**	**1/12**	0	**1/4**				
3/4	**1/12**	**1/6**	**1/6**	**1/12**	0	**-1/12**	**1/3**			
1	**1/12**	**1/6**	**1/6**	**1/12**	0	**1/6**	**-1/6**	**1/2**		
1	1/12	1/6	1/6	1/12	0	1/12	1/6	1/6	1/12	
	1/12	**1/6**	**1/6**	**1/12**	0	**1/12**	**1/6**	**1/6**	**1/12**	0

transition between these partitions, we introduce inner and outer buffer elements which use respectively the inner and outer method. The remaining elements use the base ERK method and constitute the bulk groups. Two successive bulk groups are always separated by an inner and an outer buffer element and have a time step ratio of two. Tags are introduced to distinguish the multirate groups: $3n$ for bulk, $3n+1$ for inner buffer and $3n+2$ for outer buffer, where n stands for the multirate refinement level. Figure 1 gives a simple illustrative example of the multirate groups (b) constructed from a mesh (a) with two different stable time steps. The speedup, based on the number of substantial operations, of this multirate setup compared to the singlerate version is roughly 1.4. Indeed, to reach an arbitrary final time the multirate groups 0 and 1 require twice more operations than groups 2 and 3. This method is extendable to any number of temporal refinement levels leading to multiple bulk, inner buffer and outer buffer groups.



(a) (b)

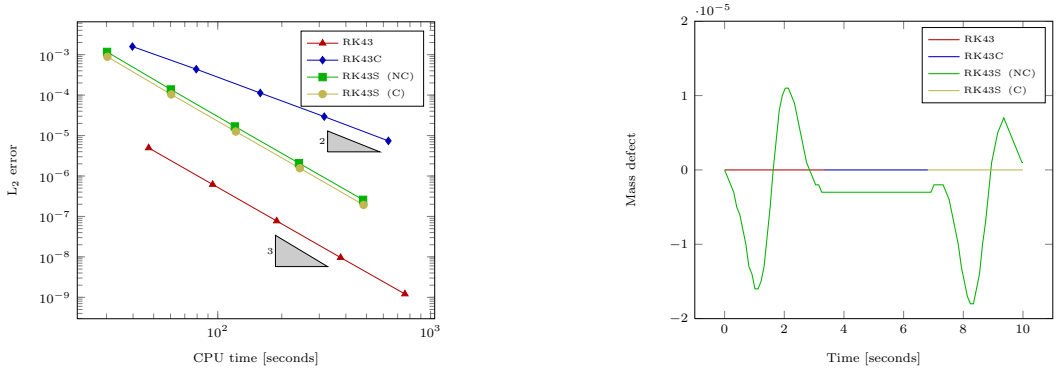
Figure 1: Element-wise stable time steps (a) and multirate groups with associated tags (b).

2.2 Convergence and conservation properties

When developing multirate schemes, particular attention should be given to the transition between the multirate groups. In particular, convergence and conservation properties should remain satisfied at each time step. The multirate methods of Constantinescu [3] are all conservative but are limited to second order accuracy in time regardless of the base ERK method. Schlegel [1] constructed a third order multirate scheme (RK43S) based on the RK43 method, that satisfies some third order conditions [2]. It remains an open question whether the additional third order conditions generally lead to third order multirate schemes. The Butcher tableau representation – characterized by a central matrix $\mathbf{A}$, a horizontal vector $\mathbf{b}$ and a vertical vector $\mathbf{c}$ – of the outer and inner buffer methods are given in Table 1(a) and 1(b). These tables have both 10 stages but, even if these stages are all required to ensure a coherent transition, the evaluation of the residuals is required at 4 stages for the outer buffer and at 8 stages for the inner buffer method (highlighted in bold). For an element-wise partitioning of the elements the method lacks conservation. Indeed, the $\mathbf{b}$ vectors of the Butcher tableaus are not identical and consequently the interface fluxes are different on both sides of the critical interface. But, by considering a partitioning by fluxes, it is possible to make the method conservative. At the end of the 10 stages, the most accurate flux (inner buffer) is imposed as unique flux at the interface. Until now and in our battery of tests, this subterfuge has never led to inconsistent results.

To illustrate these properties, a scalar advection-diffusion (constant advection and diffusion) problem is solved in a rectangular domain with an exponential initial condition and periodic boundary conditions. The spatial discretization consists of 228 $\mathrm{P}_3^{\mathrm{DG}}$ triangles of variable sizes. The element-wise stable time steps are computed with respect to the CFL condition. There are 3 refinement levels (4 bulk and 3 inner and outer buffer groups) leading to a theoretical speedup of approximatively 2. In Figure 2(a) the temporal accuracy is represented in function of the CPU time after 10 sec of physical time. The original time step is successively divided by a factor 2 such that the pure temporal error is visible. The reference method is the RK43 method. The multirate method of Constantinescu based upon this method (RK43C) is limited to second order accuracy, while the nonconservative (NC) and conservative (C) versions of the RK43S method of Schlegel are both third order accurate. But, for the same CPU time, the accuracy of the multirate methods is considerably lower compared to RK43. Indeed, the temporal error of the elements with the

largest time step propagates through the elements and becomes dominant after a significant time. However, for our applications, the temporal error is much smaller than the spatial error and the total error is expected to scale as the spatial error. The relative mass defect is shown for these methods in Figure 2(b).



(a) (b)

Figure 2: Convergence orders (a) and relative conservation defects (b).

3 A PARALLEL STRATEGY FOR MULTIRATE SCHEMES

The parallelization of this multirate scheme is challenging since the computational load is not the same at each stage of the algorithm. Likewise, the number and volume of communication at an inter-processor face depends on the multirate groups of the involved elements. Particular attention is thus given to the mesh partitioning to balance the workload among all processors at each stage of the algorithm while minimizing the inter-processor communications. Furthermore, the parallel implementation has to be considered carefully to minimize the idle times.

3.1 Multi-constraint partitioning

The number of active mesh elements at a stage of the algorithm depends on the multirate group to which they belong. Back to our example of Figure 1 and with the help of the Butcher tableaus given in Table 1, three sets of multirate groups may be identified: *subset 1* multirate tags $0, 1, 2, 3$ (active at stages 1 and 6); *subset 2* multirate tags $0, 1$ (active at stages 2,3,4,7,8 and 9); *subset 3* multirate tags $2, 3$ (active at stages 5 and 10). To ensure a perfect load balancing at each stage of the algorithm, the same amount of elements of each subset should be allocated to each partition. The mesh partitioner MeTiS allows for multiple constraints on elements and unique weights for the faces. Therefore we allocate a binary vector to each element, see Figure 3(a), that indicates whether an element belongs to a subset or not. Scalar weights are associated to each interface, see Figure3(b), representing the total communication volume between two elements. The aim is to minimize the communication to computation ratios of the processors. The partitioning strategy is very similar to the one presented in [5]. For this example and for two processors, the optimal partitioning is obtained by cutting the mesh vertically at the middle of the rectangle.



(a) (b)

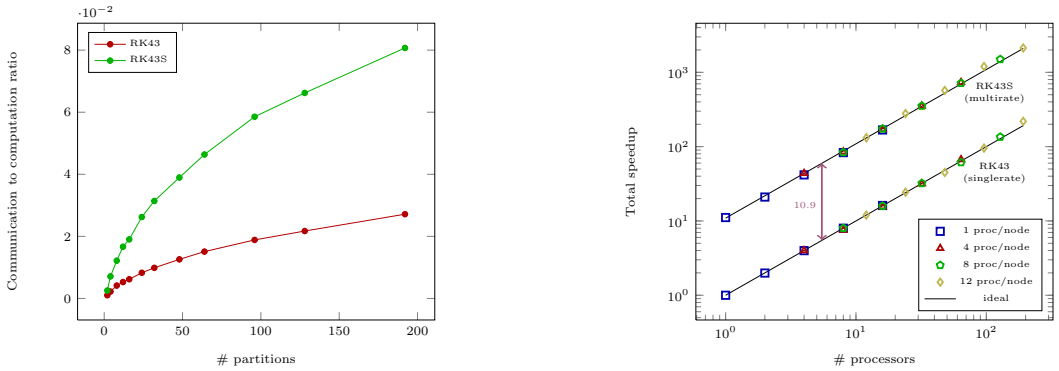
Figure 3: Associated binary vectors for elements (a) and scalar weights for faces (b) for MeTiS partitioner.

3.2 Parallel implementation

Classical MPI routines are used for the inter-processor communications in the parallel implementation. To minimize the time lost in communication, non-blocking communications are used to hide communication by computation. In the parallel implementation, multirate groups should be handled cleverly to minimize the idle times.

4 NUMERICAL RESULTS

To evaluate the quality of the mesh partitions and the global performance of our parallel multirate strategy we selected an application in the field of marine modeling. We solve the 2D shallow water equations on the sphere to simulate the tsunami that struck Japan in March 2011. The mesh is unstructured due to the complex topography and the local refinement in the region around Japan. The mesh is composed of 1,757,467 P_3^{DG} elements. A multirate setup characterized by 7 temporal refinement levels is selected and provides a theoretical speedup of roughly 10.9. An estimation of the average communication to communication ratio depending on the number of partitions is given in Figure 4(a). Note that if this estimation grows faster for RK43S, the method has a significantly lower total communication volume per time unit than RK43. The load-balancing of all the mesh partitionings is always below 1.03 (tolerance imposed to MeTiS). Strong scaling results obtained experimentally on a set of 16 computer nodes with 12 cores each are shown in Figure 4(b). Four different setups are considered depending on the number of processors used per node (1,4,8 and 12). The reference times to evaluate the total speedup for each setup correspond to the wall-clock times taken by the simulation on the considered number of processors per node with RK43 (singlerate). The RK43S method scales very well and preserves the sequential multirate speedup up to 192 processors.



(a) (b)

Figure 4: Estimation of the communication to computation ratio (a) and strong scaling (b) for RK43 and RK43S.

5 CONCLUSIONS

A third order parallel multirate strategy is proposed in the framework of DG computations. The multi-constraint partitioning strategy yields partitions of a satisfactory quality. Numerical results confirm that considerable multirate speedups are preserved up to a significant number of processors.

ACKNOWLEDGEMENT

Bruno Seny is a Research fellow with the Belgian Fund for Research in Industry and Agriculture (FRIA). Computational resources have been provided by the supercomputing facilities of the Université catholique de Louvain (CISM/UCL) and the Consortium des Equipements de Calcul Intensif en Fédération Wallonie Bruxelles (CECI) funded by the Fond de la Recherche Scientifique de Belgique (FRS-FNRS).

REFERENCES

[1] Schlegel, M., Knoth, O., Arnold, M., Wolke, R.. Multirate Runge-Kutta schemes for advection equations. *Journal of Computational and Applied Mathematics* **226**:345-357, 2009.

[2] Knoth, O., Wolke, R.. Implicit-explicit Runge-Kutta methods for computing atmospheric reactive flows. *Applied Numerical Mathematics* **28(2-4)**:327-341, 1998.

[3] Constantinescu, E.M., Sandu, A.. Multirate timestepping methods for hyperbolic conservation laws. *Journal of Scientific Computing* **33**:239-278, 2007.

[4] Seny, B., Lambrechts, J., Comblen, R., Legat, V., Remacle, J.-F.. Multirate time stepping for accelerating explicit discontinuous Galerkin computations with application to geophysical flows. *International Journal For Numerical Methods in Fluids* **71**:41-64, 2013.

[5] Seny, B., Lambrechts, J., Toulorge, T., Legat, V., Remacle, J.-F.. An efficient parallel implementation of explicit multirate runge-kutta schemes for discontinuous galerkin computations. *Journal of Computational Physics* **submitted**, 2013.