Journal of Computational and Applied Mathematics 255 (2014) 297–305

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

An enhanced parareal algorithm based on the deferred correction methods for a stiff system☆

Sunyoung Bu [a,*], June-Yub Lee [a,b]

[a] Institute of Mathematical Sciences, Ewha Womans University, Seoul 120-750, Republic of Korea

[b] Department of Mathematics, Ewha Womans University, Seoul 120-750, Republic of Korea

ARTICLE INFO

Article history:
Received 31 March 2012
Received in revised form 4 April 2013

Keywords:
Hybrid parareal algorithm
Spectral deferred correction
Krylov deferred correction
Stiff system
Differential algebraic equation

ABSTRACT

In this study, we consider a variant of the hybrid parareal algorithm based on deferred correction techniques in order to increase the convergence order even for the stiff system.

A hybrid parareal scheme introduced by Minion (2011) [20] improves the efficiency of the original parareal by utilizing a Spectral Deferred Correction (SDC) strategy for a fine propagator within the parareal iterations. In this paper, we use Krylov Deferred Correction (KDC) for a fine propagator to solve the stiff system and Differential Algebraic Equations (DAEs) stably. Also we employ a deferred correction technique based on the backward Euler method for a coarse propagator in order to make the global order of accuracy reasonably high while limiting the cost of sequential steps as small as possible.

Numerical experiments on the efficiency of our method are promising.

1. Introduction

Deferred correction methods can be used to obtain high order numerical solutions for a system of time dependent differential equations in the form of

$$f(y(t), y'(t), t) = 0, \qquad (1)$$
$$y(0) = y_0.$$

Deferred correction methods have mainly been used for boundary value problems. It has been developed by Pereyra in a series of papers [1–3] by introducing an equation for the error or defect correction to increase the accuracy of a provisional solution. In [4], the principle was applied to a ODE initial value problem. Related techniques related to the deferred and defect correction idea for initial and boundary value problems for ordinary differential equations (ODEs) were considered in [5–7]. Recently, Dutt et al. [8] introduced a new variation on the deferred/defect correction strategy called the spectral deferred correction (SDC) techniques by applying the Picard integral equation form of the correction equation and using spectral integration on Gaussian quadrature nodes [9,10]. Moreover, a new class of accelerated methods, Krylov deferred correction (KDC) methods were first studied in [11,12] to improve the accuracy of the original SDC by removing the order reduction phenomena observed in the original SDC and the divergence of the SDC methods for differential algebraic equations (DAEs). For linear problems, KDC methods use the Krylov subspace to find an optimal solution and accelerate the convergence. Further, for nonlinear problems, the spectral deferred correction procedure is used as a preconditioner for the collocation

☆ This work was supported by Priority Research Centers Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2009-0093827).

* Corresponding author. Tel.: +82 232776994.
E-mail addresses: syboo@ewha.ac.kr (S. Bu), jyllee@ewha.ac.kr (J.-Y. Lee).

0377-0427/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.cam.2013.05.001

formulation. And then the Jacobian-Free Newton–Krylov (JFNK) method [13] can be applied to the resulting preconditioned system by coupling it with the KDC technique to avoid the expensive evaluation of the Jacobian matrices [14,15].

On the other hand, the parareal algorithm, first introduced by Lions et al. [16], is a time integration scheme to compute in parallel the numerical solutions of ODEs or discretized PDEs in the temporal direction. There are three categories to parallelize time dependent differential equation systems [17–20] in general. The first approach is parallelization across the method [21,22] in which a processor is assigned to each sub-step or sub-stage of the methods, such as Runge–Kutta or linear multi-step methods. The second is categorized parallelism across the problem (or system) [23,24] in which a processor is assigned to each sub-piece of the problems by splitting of the systems, seen in techniques such as waveform relaxation. The last approach is parallelization across the steps and the parareal methods are categorized as this approach. The parareal method iteratively approximates the solution of the differential equations by assigning a processor to each time interval over the whole time domain. This parallelization scheme has received a lot of attention over the past few years. Also, the method has been studied in the context of different approaches [25], such as multi-shooting techniques or predictor–corrector methods.

A parareal algorithm, in general, can be defined by

$$y_{n+1}^k = G(t_{n+1}, t_n, y_n^k) + F(t_{n+1}, t_n, y_n^{k-1}) - G(t_{n+1}, t_n, y_n^{k-1}) \tag{2}$$

where the subscript n refers to the time subdomain number, the superscript k refers to the iteration number. F represents a fine propagator, that is, a more accurate solution on a fine grid in time interval $[t_n, t_{n+1}]$ with an initial value y_n^{k-1}. G represents a coarse propagator, a less accurate approximation in a coarser grid. Note that the F-propagator plays with the overall accuracy of the parareal method, whereas the convergence order of the method is decided by the order of the G-propagator and the number of iterations used when it is coupled with a sufficiently accurate F-propagator [20,26,27].

In this paper we are interested in increasing the order of the coarse grid propagator in the parareal method. The convergence and stability of the parareal algorithm has been studied in [25,26,28]. The convergence analysis in [25,27] using an exact F-propagator and m-th order method G-propagator gives the following estimation result

$$\|y(t_{n+1}) - y_{n+1}^k\| \leq C_k \Delta t^{m(k+1)} \tag{3}$$

where C_k is an appropriate convergence constant at the k-th iteration, and Δt is the desired time step in the coarse grid. There are two possibilities to increase the overall convergence order mk of the method: (1) one is to increase the order m of G-propagator and (2) the other is to adjust the number of the parareal iteration k. Boosting the order of the G-propagator is more efficient to obtain a solution with the desired convergence order in the parareal scheme since the cost of increasing the order of the G-propagator is much cheaper than using highly accurate F-propagator.

There are several ways to raise the convergence order of the G-propagator. A theoretical analysis of the parareal algorithm shows that the stability of the method depends on the choice of G-propagator [26,28]. Especially, for highly stiff systems or DAEs, the G-propagator is required to satisfy the L-stability property [28]. Higher order methods such as the trapezoidal rule or Runge–Kutta methods are first choices for high order G-propagator but they need to satisfy restricted stability conditions. In addition, a linear multistep method satisfying the A-stable condition can be another choice, but the highest order of accuracy for such a method is only 2. In this paper, to increase the order of these methods, we employ a deferred correction technique for the G-propagator instead of using other existing methods. Especially, Deferred Correction (DC) based on Backward Euler (BE) method is used for the G-propagator, since BE is unconditionally stable and it has L-stability [29,30], where it can unconditionally fulfill the stability condition of the parareal methods.

This paper is organized as follows. In Section 2, we briefly describe the original parareal technique and its stability and convergence properties. Also a hybrid parareal method using SDC for fine propagator is described in the section. In Section 3, we propose an enhanced parareal algorithm which uses KDC for fine propagator and a deferred correction technique combined with a lower order method for coarse propagators. In Section 4, preliminary numerical results are presented to compare the new scheme with existing ones. Finally in Section 5, we summarize our results and discuss possibility to improve efficiency of the new scheme.

2. The parareal method and a hybrid parareal method

2.1. The parareal algorithm

As in the general parareal algorithm, we assume the time interval $[0, T]$ is divided into N_p intervals with each interval being assigned to a different processor, denoted by P_0 through P_{N_p-1}. At each interval, the parareal method iteratively computes a succession of approximations $y_{n+1}^k \approx y(t_{n+1})$, where k denotes the iteration number.

Each iteration step of the parareal method is defined using two propagation operators $G(t_{n+1}, t_n, y_n)$ and $F(t_{n+1}, t_n, y_n)$. The $G(t_{n+1}, t_n, y_n)$ operator provides rough approximations of $y(t_{n+1})$, the solutions of Eq. (1) with a given initial condition, y_n, whereas the $F(t_{n+1}, t_n, y_n)$ operator typically gives a highly accurate approximation of $y(t_{n+1})$. G is usually a lower order method computed on a much coarser grid, while F is a higher-order method on a finer grid. The goal of the parareal method is to approximate a solution of F applied in serial by combining solutions of the G and F-propagators.

The parareal method begins by sequentially computing y_n^0, using the G-propagator,

$$y_{n+1}^0 = G(t_{n+1}, t_n, y_n^0) \quad \text{for } n = 0, \ldots, N_p - 1. \tag{4}$$

Once each processor has a value y_n^k with initially $k = 0$, the processors can compute the approximation in parallel

$$F_{n+1}^k = F(t_{n+1}, t_n, y_n^k) \quad \text{for } n = 0, \ldots, N_p - 1. \tag{5}$$

The parareal algorithm then computes the serial correction step,

$$y_{n+1}^{k+1} = G(t_{n+1}, t_n, y_n^{k+1}) + F_{n+1}^k - G_{n+1}^k \quad \text{for } n = 0, \ldots, N_p - 1 \tag{6}$$

where $G_{n+1}^k = G(t_{n+1}, t_n, y_n^k)$ are values saved in the previous iteration step. The iterative method proceeds alternating between the parallel computation of F_{n+1}^k in (5) and the serial computation y_{n+1}^{k+1} in (6).

2.2. Stability analysis for the parareal algorithm

The stability of parareal methods has already been studied in [25,26,28]. Consider a constant coefficient ODE with non-positive constant λ,

$$y' = \lambda y, \qquad y(0) = y_0. \tag{7}$$

Let $r = r(\lambda \delta t)$ be the stability function for the fine F-propagator using time step size δt and $R = R(\lambda \Delta T)$ is for the coarse G-propagator using time step size ΔT. According to [28], the iterative parareal algorithm is stable as long as

$$\frac{\bar{r} - 1}{2} \leq R \leq \frac{\bar{r} + 1}{2} \tag{8}$$

where $\bar{r} = r((\lambda \delta t)^s)$ and $s = \frac{\Delta T}{\delta t}$. Especially, for stiff systems, stability is guaranteed when the following condition is satisfied:

$$R_\infty = \lim_{z \to -\infty} |R(z)| \leq \frac{1}{2} \tag{9}$$

with the F-propagator as nearly accurate as the exact solver. As a consequence, the backward Euler method ($R_\infty = 0$) is unconditionally stable, and the trapezoidal rule ($R_\infty = 1$) is unstable unless the number of the overall iterations approaches to the number of intervals.

2.3. A hybrid parareal method based on SDC

Since any numerical method for the F and G-propagators can be used in the parareal techniques, it is not surprising that SDC techniques can be embedded into the parareal framework. In [20,31], a hybrid parareal spectral deferred correction method was introduced in which SDC strategies are utilized within the parareal iteration for the F-propagator. Instead of directly using the SDC scheme requiring several iterations (SDC sweeps) in serial, it is efficient for the F-propagator in each parareal iteration to perform one or a few SDC sweeps on the solution from the previous parareal iteration. As the parareal iterations converge, the F solution still converges to the high-accuracy SDC solution.

The advantage of this scheme is that the F-propagator becomes much cheaper than a full accurate solver by combining the parareal iterations and SDC iterations. Typically, the original efficiency is bounded by $1/K$, where K is the number of iterations for the parallel iterations to converge. However, the parallel efficiency of the hybrid parareal SDC is about K_s/K, where K_s is the number of iterations required of the serial SDC method to converge to a given tolerance. The hybrid parareal SDC method is a reasonably good choice for non-stiff systems.

3. An enhanced parareal algorithm based on deferred corrections

3.1. Parareal algorithm with Krylov deferred correction for the F-propagator

Both SDC and KDC methods can be used for accelerating the convergence of time marching schemes. However, SDC methods applied to stiff ODEs show order reduction phenomena unless time steps size is small enough [11], and the methods for many DAE systems become divergent independent of the time step-size selection [12]. Therefore, KDC method is more appropriated choice as an F-propagator in our parareal scheme to handle a stiff system and DAEs.

In the KDC methods, a Picard type collocation formulation is preconditioned using low-order time integration schemes based on SDC, and the resulting system is solved efficiently using a Newton–Krylov (NK) method where the optimal solution is sought for in the Krylov subspace instead of simple iterative refinements as seen in SDC iterations (or known as sweeps). Notice that Newton–Krylov method is designed for solving nonlinear algebraic equations of the form $M(x) = 0$ and is a

nested iterative method. NK methods for non-linear system consist of two different iterations, outer NK iterations and inner Krylov iterations. In the outer iteration, Newton type methods [32] are applied to linearize the nonlinear system. In the inner iterations, Krylov subspace methods [33] are employed to find solution by reducing the residual of the linearized equation by a prescribed factor and then restart the outer Newton iterations.

Using full outer NK iterations for each parareal iteration would be inefficient as discussed in hybrid parareal SDC schemes [20,31]. Instead of using full NK iterations needed for desired accuracy, we consider using one or two NK iterations per parareal iteration for the F-propagator.

3.2. Deferred correction scheme based on backward Euler for the G-propagator

For the numerical solution, we introduce a grid $t_n = n\Delta t$ in time, where Δt is the time step, and we let y_n denote the approximation of $y(t_n)$. The Backward Euler (BE) is written as

$$\frac{y_{n+1} - y_n}{\Delta t} = f(y_{n+1}, t_{n+1}). \tag{10}$$

Introducing the forward and backward difference operators defined in [29,34,35], $D_+y_n = \frac{y_{n+1}-y_n}{\Delta t}$, $D_-y_{n+1} = \frac{y_{n+1}-y_n}{\Delta t}$, the BE can be rewritten as

$$D_-y_{n+1}^{1st} = f(y_{n+1}^{1st}, t_{n+1}). \tag{11}$$

We calculate the local truncation error of the BE scheme by a Taylor expansion,

$$e(y, t_{n+1}, \Delta t) = \frac{y_{n+1} - y_n}{\Delta t} - f(y_{n+1}, t_{n+1}) \tag{12}$$

$$= -\frac{1}{2}\Delta t y''(t_{n+1}) + O(\Delta t^2). \tag{13}$$

The 2nd order approximation for BE can be achieved by solving

$$D_-y_{n+1}^{2nd} = f(y_{n+1}^{2nd}, t_{n+1}) - \frac{1}{2}\Delta t D_+D_+y_n^{2nd} + O(\Delta t^2). \tag{14}$$

We apply a predictor–corrector scheme for this implicit Eq. (14) by replacing $D_+D_+y_n^{2nd}$ with a known value $D_+D_+y_n^{1st}$, which gives a deferred correction technique as follows:

$$D_-y_{n+1}^{2nd} = f(y_{n+1}^{2nd}, t_{n+1}) - \frac{1}{2}\Delta t D_+D_+y_n^{1st} + O(\Delta t^2) \tag{15}$$

where y_{n+1}^{1st} is calculated with (11).

For another increment of the convergence order, the deferred correction algorithm can be written

$$D_-y_{n+1}^{p_{th}} = f(y_{n+1}^{p_{th}}, t_{n+1}) + \sum_{i=1}^{p-1} c_i(\Delta t)^i(D_+)^{i+1}y_n^{p-1_{th}} + O(\Delta t^p) \tag{16}$$

where $y_n^{p_{th}}$ denotes a p-th order approximation of $y(t_n)$ and c_k is a coefficient determined by the local truncation error of the scheme.

For more details of the deferred correction scheme and its stability and error estimates, the readers are referred to [29,34–36].

3.3. Informal description of the method

We assume the time interval of interest $[0, T]$ is divided into N_p uniform intervals, and each interval $[t_n, t_{n+1}]$ is assigned to a corresponding processor P_n. Note that y_n^k denotes the approximation after the k-th parareal iteration at the node point t_n. For the G-propagator, we use the deferred correction scheme based on BE as discussed in Section 3.2 and the KDC method is employed for the F-propagator in a similar fashion to the hybrid parareal SDC methods [20,31]. As seen in [20,31], only one or two outer KDC iterations are used in each parareal iteration for efficiency.

Predictor Step Initialization ($k = 0$)

Starting with the initial value y_0 on Processor P_0, compute the initial approximation y_{n+1}^0 as follows:

- Compute y_{n+1}^0 by applying the Deferred Correction (DC) technique based on backward Euler

$$y_{n+1}^0 = G(t_{n+1}, t_n, y_n^0) \tag{17}$$

sequentially for $n = 0, \ldots, N_p - 1$ starting with $y_0^0 = y_0$.

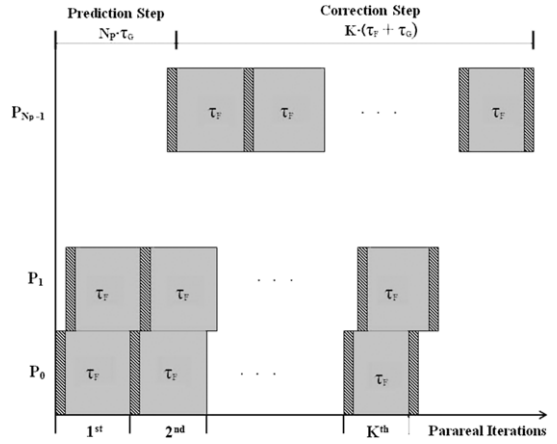

Fig. 1. Total time cost of the KDC method in parallel mode.

Corrector Step Parareal Iteration (for $k = 0, \ldots, N_p - 1$)

1. Using higher order KDC techniques with one or two KDC outer iterations, compute the approximation
$$F_{n+1}^k = F(t_{n+1}, t_n, y_n^k) \tag{18}$$
on the fine grid in parallel.
2. Receive the new initial value y_n^{k+1} from the previous processor P_{n-1} and apply the Deferred Correction technique for $G(t_{n+1}, t_n, y_n^{k+1})$.
3. The parareal algorithm approximates the solution y_{n+1}^{k+1} on processor P_n as follows and sends it to the next processor for the new initial condition.
$$y_{n+1}^{k+1} = G(t_{n+1}, t_n, y_n^{k+1}) + F_{n+1}^k - G_{n+1}^k \tag{19}$$
where $G_{n+1}^k = G(t_{n+1}, t_n, y_n^k)$ are saved value from the previous iteration.

3.4. Overall efficiency

Work in [20] presents an analysis of the theoretical parallel speedup and efficiency of the hybrid parareal/SDC method. Since the enhanced parareal KDC algorithm have quite a similar structure to the hybrid parareal/SDC method, we present the theoretical parallel efficiency for the enhanced parareal methods using the same terminology seen in [20].

To begin with, we assume that each processor is identical and the communication time among processors is negligible. The time costs for a processor to compute one step of the numerical method used for the G-propagator and F-propagator are denoted by τ_G and τ_F, respectively. Also, we denote the number of processors by N_P.

To investigate the speedup or efficiency of the enhanced parareal method, first we consider the total cost of the serial KDC with full iterations. Since the enhanced parareal KDC method starts with the G-propagator in a serial manner, the methods incur $N_P\tau_G$ time costs as seen in Fig. 1. In addition to this, the cost of each parareal iteration is $\tau_F + \tau_G$, so the cost of K parareal iterations is $K(\tau_F + \tau_G)$. Therefore, the total wall-clock cost of the enhanced parareal KDC method is $N_P\tau_G + K(\tau_F + \tau_G)$.

Based on the analysis above, the speedup of the method S is approximately

$$S \simeq N_P \cdot \frac{M\tau_F}{N_P\tau_G + K(\tau_F + \tau_G)} \tag{20}$$

where M is the number of KDC iterations needed to compute the desired accurate solution in serial. Note that M is typically smaller than K. If the parallel efficiency E using N_P processors is defined as S/N_P, then

$$E \simeq \frac{M\tau_F}{N_P\tau_G + K(\tau_F + \tau_G)}. \tag{21}$$

Let $\alpha = \frac{\tau_G}{\tau_F}$, we can rewrite Eq. (21)

$$E \simeq \frac{M}{\alpha N_P + K(1+\alpha)} = \frac{M}{\alpha(N_P + K) + K}. \tag{22}$$

We can achieve full parareal efficiency M/K when $\alpha(N_P + K)$ is much smaller than K/M. However, when N_P is large for long-term simulations or α is not negligibly small, it is not possible to get full efficiency. In practice, the most important parameter is K/M so that it is desirable to have a parareal scheme whose $K/M(\geq 1)$ ratio is as small as possible. For more details on efficiency, readers may refer to [20].

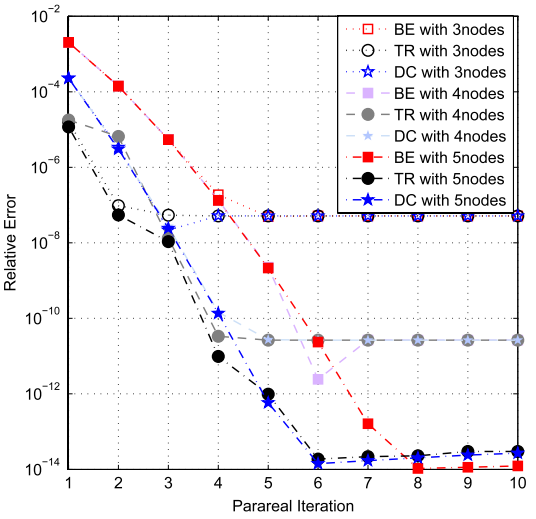

Fig. 2. Convergence of the parareal iterations using different G-propagators and accuracy limit depending on F-propagators using 3–5 KDC node points.

4. Numerical results

In this section, preliminary numerical results are presented to examine the convergence behavior and the efficiency of the new scheme compared to the standard implementation of the parareal schemes. Fig. 2 in example 4.1 shows the accuracy limit depending on the choice of F-propagator. Figs. 2 and 3 in example 4.2 show the order of convergence by choosing BE/TR/DC method as a G-propagator. Also example 4.2 demonstrates that computational cost of DC based on BE to get the desired accuracy is cheaper than the BE-based parareal method and a serial KDC method. Finally example 4.3 numerically proves that our method is stable for the stiff system while TR-based one becomes unstable.

A parareal algorithm in general shows significant speed-up for a large system which is main target of the method, however, we provide only simple test sets to validate the order of convergence and the stability property of the proposed enhanced parareal method. Applying this scheme to very large systems such as power system requires more works and complicated analysis and we are planning to present results along these directions in near future.

4.1. A simple DAE

In the first example, we consider a simple DAE system,

$$y_1' = -2y_1 + 3\mathrm{e}^{-4t} \tag{23}$$

$$y_2' = -y_1(y_2 + \sin(t)) - y_3 \tag{24}$$

$$0 = y_1 + \sin(t) + y_3 - \cos(t) \tag{25}$$

where an analytic solution is $[2.5\mathrm{e}^{-2t} - 1.5\mathrm{e}^{-4t}, -\sin(t), \cos(t)]$. To understand the convergence behaviors of the enhanced parareal method, three choices are considered for the G-propagators: the first order Backward Euler (BE) method, Trapezoidal Rule (TR) and the second order Deferred correction (DC) method based on BE for the G-propagator. KDC methods for the F-propagator using a different number of node points are also considered for the comparison. For the numerical computation, we march from $t_0 = 0.0$ to $t_F = 1.0$ with 10 processors, i.e, the time stepsize is $\Delta t = 0.1$.

In Fig. 2, the error is plotted as a function of parareal iterations for different G-propagators (BE, TR, and DC) and different Radau IIa nodes (3–5 nodes) in the F-propagator. Note that the KDC methods using p Radau IIa nodes is converging with an approximate order of $2p - 1$ [11,12]. Therefore the KDC methods with 5 node points have higher accuracy than with 3 or 4 nodes. Fig. 2 validates the fact that the overall convergence behavior depends on the choice of G-propagator, whereas the accuracy of the converged solution depends on the number of Radau IIa nodes in the KDC methods.

4.2. A nonlinear DAE

In the second example, we consider a simple nonlinear index 1 DAE system,

$$y_1' = -4y_2y_3 \tag{26}$$

$$y_2' = y_3 \tag{27}$$

$$0 = y_1 + y_2^2 - y_3^2 \tag{28}$$

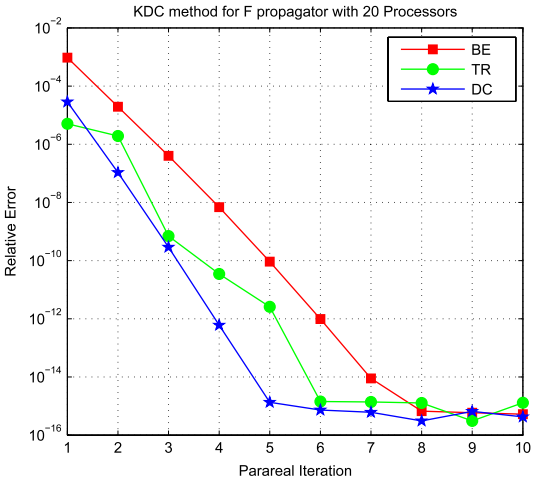

Fig. 3. Convergence of parareal iterations using different G-propagator.

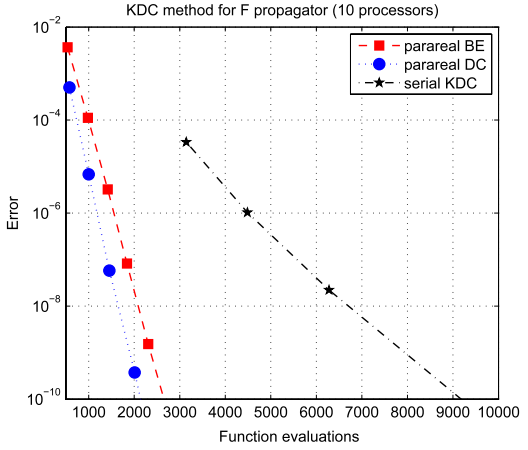

Fig. 4. Convergence of parareal iterations using different G-propagator.

where an analytic solution is $[\cos(2t), \sin(t), \cos(t)]$. To understand the convergence behaviors of the new scheme for the choice of G-propagators, the first order Backward Euler (BE) methods, second order Trapezoidal Rule (TR) and the second order Deferred correction (DC) based on BE are considered for this comparison. For the comparison, we march from $t_0 = 0.0$ to $t_F = 1.0$ with 20 processors, i.e, the time stepsize is $\Delta t = 0.05$. For the F-propagator, the KDC method with 5 Radau II nodes is employed, and each parareal iteration performs the 2 outer Newton iterations for desired efficiency and the other conditions such as the tolerance for the Newton–Krylov methods or nonlinear solvers are fixed for all simulations.

In Fig. 3, we plot the error at the final time ($t = 1.0$) versus the parareal iterations for different G-propagators (BE, TR and DC). It can be seen that after a certain number of parareal iterations, the error levels reach a certain tolerance level regardless of the G-propagator. However, the convergence behaviors are different in terms of the choice of G-propagator: it shows that higher order methods (TR, and DC) have faster convergence than BE. Thus, this indicates the higher order methods for the G-propagator can increase the rate of convergence of the parareal scheme.

We next investigate the efficiency of the enhanced parareal method by plotting the error versus total time costs for serial KDC and the parareal method. For our simulations, BE and DC are used for the G-propagator in the parareal scheme and 5 Radau IIa node points for the KDC methods are used for the F-propagator. A serial KDC method to march from $t_0 = 0.0$ to $t_F = 1.0$ with 10 processors. In Fig. 4, it can be seen that the parareal technique based on both low-order methods has a better parallel speedup as compared to the serial KDC methods. By the analysis in Eq. (21) for the backward Euler based parareal scheme, the theoretical efficiency to get 10 digit accuracy is calculated as

$$E \simeq \frac{M\tau_F}{N_P\tau_G + K(\tau_F + \tau_G)} \simeq \frac{9200}{2500} \simeq 4. \tag{29}$$

Also, in Fig. 4, we further investigate how the error changes as function evaluations using two different low-order methods for the G-propagators. It can be seen that the error behaviors of the two different methods (BE and DC) are similar, while

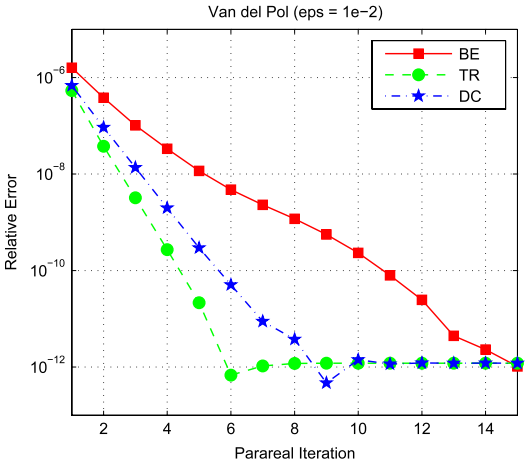

Fig. 5. Convergence of parareal iterations for the non-stiff system, $\epsilon = 10^{-2}$.

the parareal scheme based on the deferred corrections (DC) needs fewer function evaluations and hence is more efficient for the same accuracy requirements.

4.3. Van der Pol oscillator

In our third example, we consider the Van der Pol oscillator [37], which after rescaling gives

$$\begin{aligned} y_1' &= y_2 \\ y_2' &= (-y_1 + (1 - y_1^2)y_2)/\epsilon \end{aligned} \tag{30}$$

with the initial values $[y_1(0), y_2(0)] = [2, -0.66666654321]$. This is a popular nonlinear stiff ODE test problem, it describes the behavior of the vacuum tube circuits [37]. As ϵ approaches zero, the system becomes stiff. In this experiment, to test for the non-stiff case, we set $\epsilon = 10^{-2}$ and use 5 Radau IIa node points in each time step for the F-propagator and march from $t_0 = 0.0$ to $t_F = 0.1$ with 20 processors, i.e, $\Delta t = 0.005$. To compare the accuracy levels, the relative L^2 error is calculated at the final time compared to the reference solution computed by Runge–Kutta methods with an adaptive time step size and a tolerance of 10^{-11}. Fig. 5 shows that for a non-stiff case, the convergence behavior in terms of the parareal iterations depends on the convergence order of the G-propagator as we observed in previous examples.

To investigate the convergence behavior of the stiff system, in Fig. 6, we choose ϵ from a range of 10^{-3}–10^{-7} and use 5 Radau IIa node points, as above. We plot how the error changes as a function of the degree of stiffness ($\epsilon = 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}$ and 10^{-7}) using BE, DC and TR for the G-propagator when marching from $t_0 = 0.8$ to $t_F = 0.806$ with 60 processors. Fig. 6(a) and (b) show that for stiff systems, both the new method DC and BE converge to a certain accuracy level, while the second order DC has faster convergence than the first order BE. However, for stiff systems, in the parareal framework the trapezoidal rule TR is not appropriate for the G-propagator. As discussed above, the stability region of the parareal scheme is related to the stability region of the G- and F-propagators. Especially, for highly stiff system, the G-propagator should be L-stable. Fig. 6(c) shows that the parareal schemes based on TR for the G-propagator are not convergent and the error after few iterations diverges as the system gets stiffer.

5. Conclusions and discussions

In this paper, deferred correction methods are examined to boost the order of the parareal algorithm, especially for stiff systems. The stability of the parareal scheme for stiff systems is guaranteed by constructing an enhanced parareal algorithm with the L-stability method for G-propagator and KDC method for F-propagator. Analysis and numerical experiments show that the deferred correction method based on backward Euler gives faster convergence for parareal techniques by increasing the order of the parareal methods. Notice that the number of function evaluations for the deferred correction methods is bigger than that of one-step methods such as BE, however, the cost difference between different G-propagators is negligible since the cost of the F-propagator dominates the total cost.

Currently, we are investigating on the accuracy behavior of different combinations of G-propagator and F-propagator in order to improve the overall efficiency of the parareal iterations. One possibility is to use higher order but less expensive schemes for G-propagator. One of ongoing research is to apply higher order schemes such as KDC methods for G-propagator by coupling multigrid based methods such as Full Approximate Scheme (FAS) to reduce computationally cost.

Other direction of our research is how to select adaptive step-size for G- and F-propagator. There are several theoretical analysis using the constant step size [21,26,28] but the work for adaptive mesh requires more complicated analysis (refer

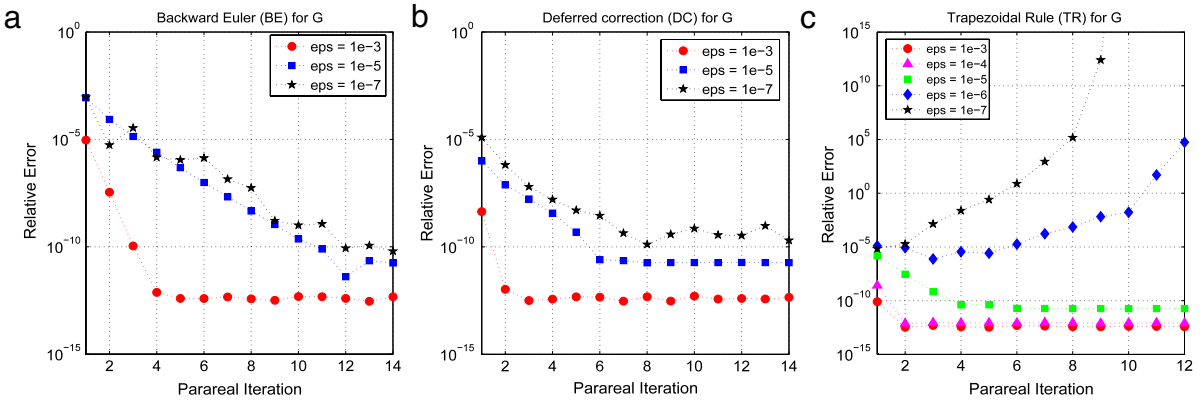

Fig. 6. Convergence of parareal iterations for the stiff system.

to [21,38,39]). Even though we used uniform grid for time marching just for simplicity, we are currently working on implementation and analysis of a fully adaptive scheme and the results will be reported soon.

References

[1] V. Pereyra, Iterated deferred correction for nonlinear operator equations, Numer. Math. 10 (1967) 316–323.
[2] V. Pereyra, Iterated deferred correction for nonlinear boundary value problems, Numer. Math. 11 (1968) 111–125.
[3] V. Pereyra, Highly accurate numerical solution of quasilinear elliptic boundary value problems in n dimensions, Math. Comp. 11 (1970) 771–783.
[4] J. Daniel, V. Pereyra, L. Schumaker, Iterated deferred corrections for initial value problems, Acta Cient. Venezolana 19 (1968).
[5] M.V. Daele, T.V. Hecke, G.V. Berghe, H.D. Meyer, Deferred correction with mono-implicit Runge–Kutta methods for first-order IVPs, J. Comput. Appl. Math. (1999).
[6] H.B. Keller, V. Pereyra, Difference methods and deferred corrections for ordinary boundary value problems, SIAM J. Numer. Anal. 16 (2) (1979).
[7] P.E. Zadunaisky, On the estimation of errors propagated in the numerical integration of ordinary differential equations, Numer. Math. 27 (1976) 21–40.
[8] A. Dutt, L. Greengard, V. Rokhlin, Spectral deferred correction methods for ordinary differential equations, BIT 40 (2) (2000) 241–266.
[9] D. Gottlieb, S.S. Orszag, Numerical Analysis of Spectral Methods, SIAM, Philadelphia, 1977.
[10] L. Greengard, Spectral integration and two-point boundary value problems, SIAM J. Numer. Anal. 28 (1991) 1071–1080.
[11] J. Huang, J. Jia, M.L. Minion, Accelerating the convergence of spectral deferred correction methods, J. Comput. Phys. 214 (2) (2006) 633–656.
[12] J. Huang, J. Jia, M.L. Minion, Arbitrary order Krylov deferred correction methods for differential algebraic equations, Comput. Phys. 221 (2) (2007) 739–760.
[13] D.A. Knoll, D.E. Keyes, Jacobian-free Newton Krylov methods: a survey of approaches and applications, J. Comput. Phys. 193 (2004) 357–397.
[14] S. Bu, J. Huang, M.M. Minion, Semi-implicit Krylov deferred correction methods for ordinary differential equations, in: Proceedings of the American Conference on Applied Mathematics, Houston, 2009.
[15] J. Jia, J. Hunag, Krylov deferred correction accelerated method of lines transpose for parabolic problems, J. Comput. Phys. 227 (3) (2008) 1739–1753.
[16] J.J. Lions, Y. Maday, G. Turinici, A parareal in time discretization of PDE's, C. R. Acad. Sci, Paris I 332 (1) (2001) 16.
[17] P. Amodio, L. Brugnano, Parallel solution in time of ODEs: some achievements and perspectives, Appl. Numer. Math. (59) (2009).
[18] K. Burrage, Parallel and sequential methods for ordinary differential equations, in: Numerical Mathematics and Scientific Computation, in: Monographs on Numerical Analysis, Oxford Science Publications, Oxford University Press, USA, 1995.
[19] K. Burrage, Parallel methods for ODEs, Adv. Comput. Math. 7 (1–2) (1997) 1–3.
[20] M.L. Minion, A hybrid parareal spectral deferred corrections method, Commun. Appl. Math. Comput. Sci. 5 (2) (2011).
[21] J.C. Butcher, Order and stability of parallel methods for stiff problems: parallel methods for odes, Adv. Comput. Math. 7 (1997) 79–96.
[22] A. Iserles, S.P. Norsett, On the theory of parallel Runge–Kutta methods, IMA J. Numer. Anal. 10 (4) (1990) 463–488.
[23] M.L. Crow, M. Ilic, The parallel implementation of the waveform relaxation method for transient stability simulations, IEEE Trans. Power Syst. 5 (3) (1990) 922–932.
[24] S. Vandewalle, D. Roose, The parallel waveform relaxation multigrid method, in: Proceedings of the Third SIAM Conference on Parallel Processing for Scientific Computing, in: Soc. Indust. Appl. Math., 1989, pp. 152–156.
[25] M.J. Gander, S. Vandewalle, Analysis of the parareal time-parallel time-integration method, SIAM J. Sci. Comput. 29 (2) (2007).
[26] G. Bal, On the convergence and the stability of the parareal algorithm to solve partial differential equations, in: Fifteen International Conference on Domain Decomposition Methods, 2003.
[27] M.J. Gander, E. Hairer, Nonlinear convergence analysis for the parareal algorithm, in: Domain Decomposition Methods in Science and Engineering XVII, 2008, pp. 45–56.
[28] G.A. Staff, E.M. Ronquist, Stability of the parareal algorithm, in: Proceedings of the 15th International Domain Decomposition Conference, in: Lect. Notes Comput. Sci., 2003.
[29] W. Kress, Error estimates for deferred correction methods in time, Appl. Numer. Math. 57 (2007).
[30] E. Hairer, G. Wanner, Solving Ordinary Differential Equations II, Springer, 1996.
[31] M.L. Minion, S. Willimas, Parareal and spectral deferred corrections, in: AIP Conference Proceedings, Vol. 1048, 2008, pp. 388–391.
[32] C.T. Kelly, Solving Nonlinear Equations with Newton's Method, SIAM, 2003.
[33] Y. Saad, M.H. Schultz, GMRES: a generalized minimal residual algorithm for solving non-symmetric linear systems, SIAM J. Sci. Stat. Comput. 7 (1986) 856–869.
[34] B. Gustafsson, W. Kress, Deferred correction methods for initial value problems, BIT 41 (2001) 986–995.
[35] B. Gustafsson, L. Hemmingsson-Franden, Deferred correction in space and time, J. Sci. Comput. 17 (2002).
[36] W. Kress, B. Gustafsson, Deferred correction methods for initial boundary value problems, J. Sci. Comput. 17 (2002).
[37] F. Mazzia, C. Magherini, Test set for initial value problem solvers, release 2.4, Department of Mathematics, University of Bari and INdAM, Research Unit of Bari, February 2008. Available at http://www.dm.uniba.it/testset.
[38] D. Guibert, D. Tromeur-Dervout, Adaptive parareal for systems of ODEs, in: Domain Decomposition Methods in Science and Engineering XVI, in: Lecture Notes in Computational Science and Engineering, vol. 55, 2007.
[39] Y. Maday, G. Turinici, The parareal in time iterative solver: a further direction to parallel implementation, in: Proceedings of the 15th International Domain Decomposition Conference, in: Lect. Notes Comput. Sci, 2003.