ISSN 2078-3744. Вісник Львів. ун-ту. Серія мех.-мат. 2016. Випуск 82. С. 5–19
Visnyk of the Lviv Univ. Series Mech. Math. 2016. Issue 82. P. 5–19

УДК 519.6.

DEFECT CORRECTION METHODS, CLASSIC AND NEW

Winfried AUZINGER[1], **Roksolyana STOLYARCHUK**[2],
Martin TUTZ[1]

[1] *Technische Universität Wien,*
Wiedner Haupstrasse 8-10, 1040 Wien, Austria
e-mail: w.auzinger@tuwien.ac.at, martin.tutz@gmx.at

[2] *Lviv Polytechnic National University,*
12 S. Bandera Str., 79013, Lviv, Ukraine
e-mail: sroksolyana@yahoo.com

Defect correction methods are based on the idea of measuring the quality of an approximate solution to an operator equation by forming the defect, or residual, with respect to the given problem. By an appropriate backsolving procedure, an error estimate is obtained. This process can also be continued in an iterative fashion. One purpose of this overview is the further dissemination of the underlying concepts. Therefore, we first give a general and consistent review on various types defect correction methods, and its application in the context of discretization schemes for differential equations. After describing the general algorithmic templates we discuss some specific techniques used in the solution of ordinary differential equations. Moreover, new results about the application to implicit problems are presented.

Key words: defect correction, discretization, ordinary differential equations.

1. Introduction

Defect Correction (DeC) methods (also: 'deferred correction methods') are based on a particular way to estimate local or global errors, especially for differential and integral equations. The use of simple and stable integration schemes in combination with defect (residual) evaluation leads to computable error estimates and, in an iterative fashion, yields improved numerical solutions.

In the first part of this article, the underlying principle is motivated and described in a general setting, with focus on the main ideas and algorithmic templates. In the sequel, we consider its application to ordinary differential equations in more detail. The proper

2010 Mathematics Subject Classification: 65L05, 65L06.

choice of algorithmic components is not always straightforward, and we discuss some of the relevant issues.

We are not specifying all algorithmic components in detail, e.g., concerning the required interpolation and quadrature processes. But these are numerical standard procedures which are easy to understand and to realize. Also, exhaustive survey of the available literature on the topic is no provided here.

The introductory part of this text is a revised and extended version of the overview given in [1]. We motivate the DeC principle in a way slightly different from the classical paper [18], with a clear focus on the underlying error estimation principles.

In addition, some recent material is included, in particular, that concerning the role of error structures for the convergence behavior. An algorithmic version for differential equations in implicit formulation proposed in [19] is also presented.

We use upper indices for iteration counts and lower indices for numbering along discrete grids.

2. Underlying concepts and general algorithmic templates

Many iterative numerical algorithms are based on the following principle. Let an initial value y_0 be given. For $i = 0, 1, 2, \ldots$:

- Compute the residual, or 'defect', d^i of the current iterate y^i with respect to the given problem,
- backsolve for a correction ε^i using an approximate solver,
- apply the correction to obtain the next iterate $y^{i+1} := y^i - \varepsilon^i$.

Stationary iterative methods for linear systems of equations and Newton iteration for systems of nonlinear equations are classical examples. For starting our general considerations, we think of a given, *original problem* in form of a system of nonlinear equations,

$$\phi(y) = 0, \quad \text{with exact solution } \; y = y^*. \tag{1}$$

2.1. Error estimation based on nonlinear approximation.

We assume that some reasonable linear or nonlinear approximation $\tilde{\phi} \approx \phi$ is given. We consider a procedure for the purpose of estimating the error of a given approximate solution y^0 to y^*. To this end we define the *defect*

$$d^0 := \phi(y^0)$$

of y_0, i.e., the amount by which $\phi(y^0)$ fails approximate $0 = \phi(y^*)$. Furthermore, with y^0, d^0 we associate the so-called *neighboring problem* related to (1),

$$\phi(y) = d^0, \quad \text{with exact solution } \; y = y^0. \tag{2}$$

We invoke two heuristic principles, (A) and (B) in the terminology from [18], for estimating the error of y^0. Originally introduced in [18] (see also [6]), these are based on the idea that (2) may be considered to be closely related to (1), provided d^0 is small enough.

(A): Let $\tilde{y}$ and $\tilde{y}^0$ be the solutions of $\tilde{\phi}(y) = 0$ and $\tilde{\phi}(y) = d^0$, respectively; we assume that these can be formed at low computational cost. Considering original problem (1) neighboring problem (2) together with their approximations,

$$\begin{array}{ll} \phi(y^*) = 0 & \qquad \phi(y^0) = d^{\,0} \\ \tilde{\phi}(\tilde{y}) = 0 & \qquad \tilde{\phi}(\tilde{y}^0) = d^{\,0} \end{array}$$

suggests the approximate identity

$$\tilde{y}^0 - \tilde{y} \approx y^0 - y^*.$$

This leads to the

$$\text{error estimator} \quad \varepsilon^0 := \tilde{y}^0 - \tilde{y} \tag{3a}$$

as a computable estimate for the error $e^0 := y^0 - y^*$. We can use it to obtain an updated approximation y^1 in the form

$$y^1 := y^0 - \varepsilon^0 = y^0 - (\tilde{y}^{\,0} - \tilde{y}). \tag{3b}$$

(B): Consider the truncation error $\ell^* := \tilde{\phi}(y^*)$, the amount by which y^* fails to satisfy the approximate equation $\tilde{\phi}(y) = 0$. With $\tilde{d}^{\,0} := \tilde{\phi}(y^0)$, considering the approximate identity

$$\begin{aligned} &\tilde{\phi}(y^*) - \tilde{\phi}(y^0) \approx \phi(y^*) - \phi(y^0), \\ &\text{i.e.,} \quad \ell^* - \tilde{d}^{\,0} \approx -d^{\,0}, \end{aligned}$$

suggests to choose the

$$\text{truncation error estimator} \quad \lambda^0 := \tilde{d}^{\,0} - d^{\,0} \tag{4a}$$

i.e., $\lambda^0 = (\tilde{\phi} - \phi)(y^0)$, as a computable estimate for the truncation error. Note that $-d^{\,0} = \phi(y^*) - d^{\,0}$ is the truncation error of y^* with respect to (2). In the case $y^0 = \tilde{y}$, i.e., $\tilde{\phi}(y^0) = 0$, we have $\lambda^0 = -d^{\,0} \approx \ell^*$.

We can use λ^0 to obtain an updated approximation y^1 by solving

$$\tilde{\phi}(y^1) = \lambda^0, \tag{4b}$$

which also provides an estimate for the error: $\varepsilon^0 := y^0 - y^1 \approx y^0 - y^* = e^0$. Eq. (4b) can also be written in terms of the error estimate as

$$\tilde{\phi}(y^0 - \varepsilon^0) = \lambda^0, \tag{4c}$$

approximating the error equation $\tilde{\phi}(y^0 - e^0) = \ell^*$.

In general, (A) and (B) are not equivalent. However, if $\tilde{\phi}(y) = P\,y - c$ is an affine mapping, it is easy to check that (A) and (B) result in the same error estimate ε^0, which can be directly obtained as the solution of the correction equation

$$P\,\varepsilon^0 = d^{\,0}, \tag{5}$$

and the corresponding truncation error estimate is $\lambda^0 = (P\,y^0 - c) - d^{\,0}$.

2.2. **Iterated Defect Correction (IDeC).** Both approaches (A) and (B) are designed for a posteriori error estimation, and they can also be used to design iterative solution algorithms, involving updated versions of the neighboring problem in course of the iteration. This leads in straightforward way to two alternative versions the method of *Iterated Defect Correction* (IDeC), starting from an initial approximation y^0. Of course, $y^0 = \tilde{y}$ is a natural choice.

```
IDeC (A) : :  Solve $\tilde{\phi}(\tilde{y}) = 0$
              For $i = 0, 1, 2, \ldots$ :
                – Compute $d^i := \phi(y^i)$
                – Solve $\tilde{\phi}(\tilde{y}^i) = d^i$
                – Set $\varepsilon^i := \tilde{y}^i - \tilde{y}$
                – Update $y^{i+1} := y^i - \varepsilon^i$
```

The corrections ε^i play the role of successive estimates for the errors $e^i = y^i - y^*$.

```
IDeC (B) : :  Set $\lambda^{-1} := \tilde{\phi}(y^0)$
              For $i = 0, 1, 2, \ldots$ :
                – Compute $d^i := \phi(y^i)$
                – Update $\lambda^i := \lambda^{i-1} - d^i$
                – Solve $\tilde{\phi}(y^{i+1}) = \lambda^i$
```

The λ^i evolve from accumulated defects, $\lambda^i = \tilde{\phi}(y^0) - d^0 - \ldots - d^i$, playing the role of successive approximations to the truncation error $\ell^* = \tilde{\phi}(y^*)$.

An equivalent reformulation reads

```
For $i = 0, 1, 2, \ldots$ :
  – Compute $d^i := \phi(y^i)$
  – Solve $\tilde{\phi}(y^{i+1}) = (\tilde{\phi} - \phi)(y^i)$
```

Remarks.

- Nonlinear IDeC has the form of a 'full approximation scheme', where we directly solve for the new approximation in each step. If $\tilde{\phi}$ is affine, IDeC (A) and IDeC (B) are again equivalent and can be reformulated as a correction scheme in terms of linear backsolving steps for the correction $\varepsilon^i = \tilde{y}^i - \tilde{y}$, as in (5).
- IDeC (B) can also be rewritten in the spirit of (4c).
- Note that y^* is a fixed point of an IDeC iteration since $d^* := \phi(y^*) = 0$.

For systems of algebraic equations, choosing $\tilde{\phi}$ to be nonlinear is usually not very relevant from a practical point of view. Rather, such a procedure turns out to be useful in a more general context, where ϕ represents an operator between functions spaces (typically a differential or integral operator), and where $\tilde{\phi}$ is a *discretization* of ϕ. This leads us to the class of DeC methods for differential or integral equations.

3. Application to ordinary differential equations (ODEs)

We mainly focus on IDeC (A), the 'classical' IDeC method originally due to [20]. IDeC (B) can be realized in a similar way, and we will remark on this where appropriate.

3.1. **A basic version: IDeC (A) based on forward Euler.** Let us identify the original problem $\phi(y)=0$ with an initial value problem (IVP) for a system of n ODEs,

$$\tfrac{d}{dx}\,y(x)=f(x,y(x)),\quad y(x_0)=y_0, \tag{6a}$$

with exact solution $y^*(x)\in\mathbb{R}^n$. This means

$$\phi(y)(x):=\tfrac{d}{dx}\,y(x)-f(x,y(x)), \tag{6b}$$

with fixed initial condition $y(x_0)=y_0$. More precisely, the underlying function spaces and the initial condition $y(x_0)=y_0$ are part of the complete problem specification.

Furthermore, we identify the problem $\tilde\phi(y)=0$ with a discretization scheme for (6); at the moment we assume that a constant stepsize h is used, with grid points $x_l=a+l\,h,\ l=0,1,2,\ldots$. Consider for instance the first order accurate forward Euler scheme

$$\frac{Y^0_{l+1}-Y^0_l}{h}=f(x_l,{Y_l}^0),\quad l=0,1,2,\ldots, \tag{7a}$$

and associate it with the operator $\tilde\phi$ acting on continuous functions $y(x)$ satisfying the initial condition $y(x_0)=y_0$,

$$\tilde\phi(y)(x_l):=\frac{y(x_{l+1})-y(x_l)}{h}-f(x_l,y(x_l))=0. \tag{7b}$$

Choose a continuous function $y^0(x)$ interpolating the ${Y_l}^0$ at the grid points x_l. The standard choice is a continuous piecewise polynomial interpolant of degree p over $p+1$ successive grid points, i.e., piecewise interpolation over subintervals $\mathbf{I}_j$ of length ph. In the corresponding piecewise-polynomial space $\mathcal{P}_p$, $y^0(x)$ is the solution of $\tilde\phi(y)=0$. The defect $d^0:=\phi(y^0)$ is well-defined,

$$d^0(x)=\phi(y^0)(x)=\tfrac{d}{dx}\,y^0(x)-f(x,y^0(x)), \tag{8a}$$

and $y^0(x)$ is the exact solution of the neighboring IVP

$$\tfrac{d}{dx}\,y(x)=f(x,y(x))+d^0(x),\quad y(x_0)=y_0. \tag{8b}$$

We now consider a correction step $y^0\mapsto y^1$ of type (A),

$$\begin{aligned}&\text{Solve}\ \ \tilde\phi(\tilde y^0)=d^0,\\ &\text{followed by}\ \ y^1(x)\ :=\ y^0(x)-(\tilde y^0-y^0)(x).\end{aligned}$$

This means that $\tilde y^0\in\mathcal{P}_p$ is to be understood as the interpolant of the discrete values ${\tilde Y_j}^0$ obtained by the solution of

$$\frac{\tilde Y^0_{l+1}-\tilde Y^0_l}{h}=f(x_l,{\tilde Y_l}^0)+d^0(x_l),\quad l=0,1,2,\ldots,$$

which is the forward Euler approximation to (8b), with additional pointwise evaluation of the defect at the grid points x_l.

According to our general characterization of IDeC (A), this process is to be continued to obtain further iterates $y^i(x)$. If we use m IDeC steps in the first subinterval $\mathbf{I}_1=[a,a+$

$ph]$, we can restart the process at the starting point $a+ph$ of the second subinterval $\mathbf{I}_2$, with the new initial value $y(a+ph) = y^m(a+ph)$. This is called local, or active mode. Alternatively, one may integrate with forward Euler over a longer interval I encompassing several of the $\mathbf{I}_j$ and perform IDeC on I, where each individual $y^i(x)$ is forwarded over the complete interval. This is called global, or passive mode.

Remark. The exact solution y^* is not in the scope of the iteration, since the y^i live in the space $\mathcal{P}_p$. But there is a fixed point $\hat{y} \in \mathcal{P}_p$ related to y^*: It is characterized by the property $\hat{d} := \phi(\hat{y}) = 0$, i.e., $\frac{d}{dx}\,\hat{y}(x_l) = f(x_l, \hat{y}(x_l))$ for all l. This means that $\hat{y}$ is a collocation polynomial, and IDeC based on the Euler scheme can be regarded as an iterative method to approximate collocation solutions. In fact, this means that, instead of (6), the system of collocation equations $\phi(\hat{y})(x_l) = \frac{d}{dx}\,\hat{y}(x_l) - f(x, \hat{y}(x_l)) = 0$ at the collocation nodes x_l is rather to be considered as the effective original problem.

3.2. **IDeC based on higher order schemes $\tilde{\phi}$. Remarks on convergence theory.** For IDeC applied to IVPs, any basic scheme $\tilde{\phi}$ may be used instead of forward Euler. E.g., in the pioneering paper [20] a classical Runge-Kutta (RK) scheme of order 4 was used. Using RK in the correction steps means that in each individual evaluation of the right hand side the pointwise value of the current defect is to be added (RK applied to [NP]). Many other authors have also considered and analyzed IDeC versions based on RK schemes.

Despite the natural idea behind IDeC, the convergence analysis is not straightforward. Obtaining a full higher order of convergence asymptotically for $h \to 0$ requires

- a sufficiently well-behaved, smooth problem,
- a sufficiently high degree p for the local interpolants $y^i(x)$,
- sufficient smoothness of these interpolants, in the sense of boundedness of a certain number of derivatives of the $y^i(x)$, uniformly for $h \to 0$.

A typical convergence result reads as follows:

> *If the sequence of grids is equidistant and the underlying scheme has order q, then m IDeC steps result in an error $y^m(x) - y^*(x) = \mathcal{O}(h^{\min\{p,\,m\,q\}})$ for $h \to 0$, where p is the degree of interpolation.*

The achievable order p is usually identical to the approximation order of the fixed point $\hat{u} \in \mathcal{P}_p$, which corresponds to a collocation polynomial in a generalized sense.

Naturally, IDeC can also be applied to boundary value problems (BVPs). For second order two-point boundary value problems, the necessary algorithmic modifications have first been described in [9]. Here, special care has to be taken at the end points of the interpolation intervals $\mathbf{I}_j$, where an additional defect terms arises due to jumps in the derivatives of the local interpolants.

3.3. **The influence of a nonequidistant grid.** As mentioned above, the smoothness of the global error is essential for the successful performance of an IDeC iteration. A technical tool to assure the latter smoothness property are asymptotic expansions of the global discretization error $\tilde{y} - y^*$ for the underlying scheme, which have been proved to exist for RK methods over constant stepsize sequences. A convergence result for IDeC derived in this way is, e.g., given in [10]; see also [17].

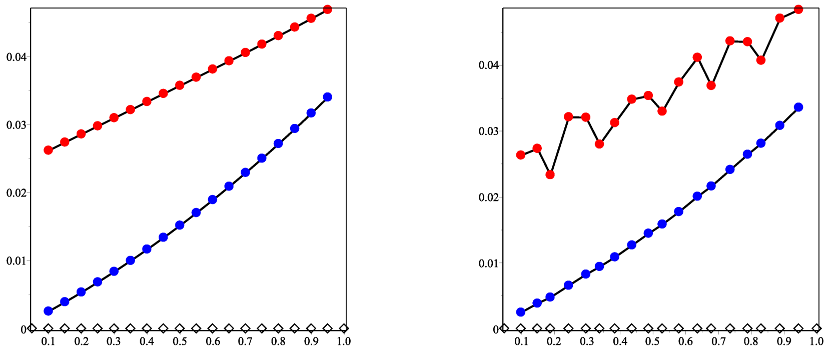

Fig. 1. Error behavior of an Euler solution, equidistant grid (left) and nonequidistant grid (right).

The assumption of a constant stepsize appears quite restrictive, but it is sufficient to assume that the stepsize h be kept fixed over each interpolation interval. We note that for IDeC algorithms, this requirement is indeed necessary, as has been demonstrated in [3]. Otherwise the error $\tilde{y} - y^*$ usually lacks the required smoothness properties, despite its asymptotic order.

To illustrate this fact, let us consider the forward Euler scheme (7a) applied to the simple ODE $y'(x) = y(x) - (\sin x + \cos x)$. For $y(0) = 1$, the solution of the IVP is $y^*(x) = \cos x$. We apply (7a) on the interval $x \in [0, 1]$ and take 20 integration steps with constant stepsize $h = 1/20$. Then we repeat the procedure on on a nonequidistant grid, where the stepsizes h_j are small relative random perturbations of the original stepsize h. Fig. 1 shows the behavior of the error (lower curve) and its first difference quotients over the grid (upper curve) for both cases. In the right plot the irregular variation of the error is clearly visible, and this effect becomes even more significant if we consider higher difference quotients. This is not difficult to explain theoretically, see [19]. As a consequence, higher derivatives of the associated interpolants $y^i(x)$ are not uniformly bounded, which would be required in the convergence theory of IDeC schemes.

3.4. Reformulation in terms of integral equations. IQDeC (A) and IQDeC (B) ('spectral IDeC').

An ODE can be transformed into an integral equation. Taking the integral means of (6a) over the interval spanned by two successive grid points gives

$$\frac{y(x_{l+1}) - y(x_l)}{h} = \int_{x_l}^{x_{l+1}} f(x, y(x))\, dx. \tag{9}$$

We observe that the left-hand side is of the same type as in the Euler approximation (7a). Therefore it appears natural to consider (9) instead of (6) as the original problem. In addition, for numerical evaluation the integral on the right-hand side has to be approximated, typically by polynomial quadratures using the $p+1$ nodes available in the current working interval $\mathbf{I}_j \ni x_l$. The coefficients depend on the location of x_l within $\mathbf{I}_j$.

Using Q as a generic symbol for these quadratures we obtain the computationally tractable, modified original problem replacing the ODE (6b), defined over the grid $\{x_l\}$

as

$$\phi(y)(x_l) := \frac{y(x_{l+1}) - y(x_l)}{h} - (Qf)(x, y(x))_l = 0, \tag{10a}$$

or, more precisely, its effective version restricted to $y \in \mathcal{P}_p$. Up to quadrature error, (10a) is an 'exact finite difference' scheme exactly satisfied by y^*. The treatment of the leading derivative term y' is the same in (10a) and in (7b), which turns out to be advantageous. (10a) leads to an alternative definition of the defect at the evaluation points x_l, namely

$$\begin{aligned} \bar{d}^i(x_l) &:= \phi(y^i)(x_l) \\ &= \frac{y^i(x_{l+1}) - y^i(x_l)}{h} - (Qf)(x, y^i(x))_l \,. \end{aligned} \tag{10b}$$

This may be interpreted in the sense that the original, pointwise defect $d^i(x)$ is 'preconditioned' by applying local quadrature. All other algorithmic components of IDeC remain unchanged, with correspondingly defined neighboring problems.

In [3], this version is introduced and denoted as IQDeC (type (A)). Variants in the spirit of IQDeC of type (B) have also been developed; this is often called 'spectral defect correction' and has first been described in [8]. For a convergence proof, see [12].

Remarks.

- With appropriate choice of defect quadrature, the fixed point of IQDeC is the same as for IDeC. In fact, the equation $\hat{d} = \phi(\hat{u}) = 0$ turns out to be closely related to a reformulation of the associated collocation equations $\hat{y}'(x_l) = f(x_l, \hat{y}(x_l))$ in the form of an exact finite difference scheme approximated via quadrature. The latter is closely related to the implicit Runge-Kutta (IRK) reformulation of the collocation equations.
- There are several motivations for considering IQDeC. The major point is that, as demonstrated in [3], its convergence properties are much less affected by irregular distribution of the x_l. This is due to the close relationship between $\tilde{\phi}$ and ϕ, see (7a) and (10a). In the forward Euler case, for instance, the normal order sequence $1, 2, 3, \ldots$ shows up, in contrast to classical IDeC.

  We also refer to [2] for a motivation and explanation of the IQDeC technique in the context of semilinear problems.
- IQDeC is also closely related to the concept of *exact difference schemes*, see, e.g., [11, 15]: Eq. (9) represents an exact difference scheme (EDS) satisfied by the true solution y^*. In the context of IQDeC, the defect is taken with respect this EDS, using an appropriate quadrature formula for evaluation of the right-hand side. However, this way of 'truncating' the EDS not the same as in [11, 15], where compact schemes are constructed and defect correction is typically not considered as an algorithmic option.

  Similar remarks apply to second order problems (which are also considered in Sec. 4 below). To our opinion, the combination of compactly truncated EDS schemes with defect correction will be worth considering as an alternative to simple fixed point iterations or more intricate Newton-like schemes applied to an EDS.

- For a related approach in the context of second order two-point boundary value problems, also permitting variable mesh spacing, see [7].
- Another modification can be used to construct superconvergent IDeC methods: In [3] ('IPDeC') and in [16], use of an equidistant basic grid is combined with defect evaluation at Gaussian nodes, in a way that the resulting iterates converge to the corresponding superconvergent fixed point (collocation at Gaussian nodes).

3.5. **Stiff and singular problems.** For stiff systems of ODEs, DeC methods have been used with some success. However, as for any other method, the convergence properties strongly depend on the problem at hand. The main difficulty for DeC is that the convergence rate may be rather poor for error components associated with stiff eigendirections. An overview and further material on this topic can be found in [4] or [8]. Similar remarks apply to problems with singularities.

3.6. **Boundary value problems (BVPs) and 'deferred correction'.** Historically, one of the first applications of a type (B) truncation error estimator (4a) appears in the context of finite-difference approximations to a BVP

$$\frac{d}{dx}\,y(x) = f(x, y(x)), \quad R(y(x_0), y(b)) = 0, \tag{11}$$

posed on an interval $[a, b]$ (with boundary conditions represented by the function R), or higher order problems. (A classical text on the topic is [14].) For a finite-difference approximation of $y'(x_l)$, e.g. as in (7a), asymptotic expansion of the truncation error ℓ^* is straightforward using Taylor series and using (11):

$$\begin{aligned} \ell^*(x_l) &= \tilde{\phi}(y^*)(x_l) = \frac{y^*(x_{l+1}) - y^*(x_l)}{h} - \frac{d}{dx}\,y^*(x_l) \\ &= \frac{1}{2}\,\frac{d^2}{dx^2}\,y^*(x_l) + \frac{1}{6}\,\frac{d^3}{dx^3}\,y^*(x_l) + \ldots \end{aligned} \tag{12}$$

The idea is to approximate the leading term $\frac{1}{2}\,\frac{d^2}{dx^2}\,y^*(x_l)$ by a second order difference quotient involving three successive nodes. This defines an approximate truncation error associated with an approximate original problem, which corresponds to a higher order discretization of (11). The corresponding estimator λ^0 is obtained by evaluating the approximate truncation error at a given $y = y^0$. This is used in the first step of an IDeC (B) procedure (see (4a)–(4c)). In this context, *updating* the (approximate) [OP] in course of the iteration is natural, involving difference approximations of the higher order terms in (12), to be successively evaluated at the iterates y^i.

IDeC (B) versions of this type are usually addressed as deferred correction techniques, and they have been extensively used, especially in the context of boundary value problems. The analysis heavily relies on the smoothness properties of the error. Piecewise equidistant meshes are usually required. A difficulty to be coped with is the fact that the difference quotients involved increase in complexity and have to be modified near the boundary and at points where the stepsize is changed.

3.7. **Defect-based error estimation and adaptivity.** In practice, the DeC principle is also applied – in the spirit of our original motivation – for estimating the error of a given numerical solution with the purpose of adapting the mesh. A typical case is described and analyzed in [5]: Assume that y^0 is a piecewise polynomial collocation solution to the BVP (11). Collocation methods are very popular and have favorable convergence properties. By definition of y^0, its pointwise defect $d^0(x) = \frac{d}{dx}\,y^0(x) - f(x, y^0(x))$ vanishes at the collocation nodes which are, e.g., chosen in the interior of the collocation subintervals $\mathbf{I}_j$. Therefore, information about the quality of y^0 is to be obtained by evaluating $d^0(x)$ at another nodes, e.g., the endpoints of the $\mathbf{I}_j$.

For estimating the global error $e^0(x) = (y^0 - y^*)(x)$ one can use the type (A) error estimator (3a) based on a low-order auxiliary scheme $\tilde{\phi}$, e.g., an Euler or box scheme, over the collocation grid. Replacing the pointwise defect d^0 by the modified defect $\bar{d}^0$, analogously as in (10b), is significantly advantageous, because this version is robust with respect to the lack of smoothness of y^0 which is only a C^1 function. In [5] it has been proved that such a procedure leads to a reliable and asymptotically correct error estimator of QDeC type.

With an appropriately modified version of $\bar{d}^0$, closely related to the defect definition from [7], the QDeC estimator can also be extended to second (or higher order) problems.

4. Extensions

In this section we describe recent extensions of the I*DeC technique (version A) to regular implicit first and second order initial value problems in more detail. Numerical results for selected test examples are also presented. Clearly, these versions can also be applied to the special case of explicit ODEs.

4.1. IQDeC (A) for implicit first order ODEs – IIQDeC.

Consider a first order initial value problem of the type

$$F\Big(x, y(x), \tfrac{d}{dx}\,y(x)\Big) = 0, \quad y(x_0) = y_0. \tag{13}$$

The IIQDeC algorithm for the solution of (13) is an extension of the IQDeC approach explained in Sec. 3.4. For the numerical solution of (13) we introduce a grid comprising several subintervals $\mathbf{I}_1, \mathbf{I}_2, \ldots$, where the relative position of the grid points within the $\mathbf{I}_j$ is determined by $m+1$ parameters $0 \leqslant c_0 < c_1 < \ldots < c_{m-1} < c_m \leqslant 1$, and the absolute position is given by

$$x_{j,l} = \mathbf{x}_{j-1} + c_l\,\mathbf{h}_j, \quad j = 1, 2, \ldots, \quad l = 0 \ldots m,$$

where $\mathbf{h}_j$ denotes the length of the subinterval $\mathbf{I}_j$. On this grid, a first approximation $Y^0_{j,l}$, using the backward Euler scheme as basic discretization, is computed, i.e., we solve

$$F\Big(x_{j,l}, Y^0_{j,l}, \tfrac{Y^0_{j,l} - Y^0_{j,l-1}}{h_{j,l}}\Big) = 0, \tag{14}$$

starting from $Y^0_{1,0} = y_0$ at $x_{1,0} = x_0$. The $Y^0_{j,l}$ and, later on, the $Y^i_{j,l}$ $(i = 1, 2, \ldots)$ are interpolated by polynomials $p^i_j(x)$ of degree $\leqslant m$, which define the piecewise polynomial function $p^i(x) = p^i_j(x)$. The pointwise defect of $p^i_j(x)$ with respect to (13) is given by

$$d^i_j(x) = F\Big(x, p^i_j(x), \tfrac{d}{dx}\,p^i_j(x)\Big), \quad x \in \mathbf{I}_j.$$

Now we define the locally integrated defect, an extension of (10b) to the implicit case, by

$$\bar{d}^i_{j,l} := \sum_{\mu=1}^{m} \alpha_{l,\mu}\, d^i_j(x_{j,\mu}) \approx \frac{\displaystyle\int_{x_{j,l-1}}^{x_{j,l}} d^i_j(x)\, dx}{x_{j,l} - x_{j,l-1}}\,. \tag{15}$$

The $\alpha_{l,\mu}$ are the weights of the corresponding interpolatory quadrature formulas with nodes $c_1, \dots, c_m$ and degree of exactness $m-1$. With the basic discretization scheme (14), and the defect (15), the discretized neighboring problem reads

$$F\Big(x_{j,l}, \tilde{Y}^i_{j,l}, \tfrac{\tilde{Y}^i_{j,l}-\tilde{Y}^i_{j,l-1}}{h_{j,l}}\Big) = \bar{d}^i_{j,l}, \tag{16}$$

starting from $\tilde{Y}^i_{1,0} = y_0$ at $x_{1,0} = x_0$. With the solution of (16), the improved approximations are defined by

$$Y^i_{j,l} = Y^0_{j,l} - \big(\tilde{Y}^{i-1}_{j,l} - Y^{i-1}_{j,l}\big), \quad i = 1, 2, \dots. \tag{17}$$

For a convergence proof of the IIQDeC method, see [19].

Example 1. Consider the implicit scalar nonlinear test problem

$$\begin{aligned} e^{y'(x)} + y'(x) + y(x) & \qquad\qquad (18a)\\ = e^{-\sin x} + \cos x - \sin x, &\\ y(0) = 1, & \qquad\qquad (18b) \end{aligned}$$

with exact solution $y^*(x) = \cos x$. The numerical solution is computed over a sequence of subintervals of length **h**, each of them divided into a nonequidistant grid with 4 'randomly' chosen nodes ($c_1 = 0.1234$, $c_2 = 0.5054$, $c_3 = 0.7134$, $c_4 = 1$), in order to demonstrate the robustness of IQDeC with respect to varying stepsizes.

We choose the integration interval $x \in [0, 3]$. The resulting global errors with respect to the exact solution at the endpoint $x = 3$ are displayed in Table 1 together with the observed convergence orders. Results are given for the basic scheme (BEUL), 4 IIQDeC iterates working in passive mode, and the fixed point of the IIQDeC iteration (COLL, corresponding to collocation at the points where the defect is evaluated).

4.2. IPDeC (A) for implicit second order ODEs – IIPDeC2-DQ2.

Here we present a new superconvergent I*DeC algorithm for implicit initial value problems of second order. Consider a problem of the type

$$F\Big(x, y(x), \tfrac{d}{dx}\, y(x), \tfrac{d^2}{dx^2}\, y(x)\Big) = 0, \tag{19a}$$

$$y(x_0) = y_0, \ \tfrac{d}{dx}\, y(x_0) = y_0'. \tag{19b}$$

The IIPDeC 2 algorithm for the solution of (19) is an extension of the IPDeC approach mentioned at the end of Sec. 3.4. It is based on a combination of an equidistant grid $\{x_{j,l},\ l = 0 \dots m\}$ with constant inner stepsize h in each interval $\mathbf{I}_j$, and another grid $\{\hat{x}_{j,k},\ k = 1 \dots \hat{m}\}$ ($\hat{m} = m-1$) based on Lobatto nodes (with parameters $\hat{c}_k$).

h	BEUL	IIQDeC/1	IIQDeC/2	IIQDeC/3	COLL
0.1	6.31E-03	1.14E-04	1.02E-06	3.83E-09	3.98E-09
0.05	3.16 E-03	2.90E-05	1.31E-07	2.69E-10	2.43E-10
0.025	1.58E-03	7.30E-05	1.66E-08	1.77E-11	1.50E-11
0.0125	7.91E-04	1.83E-06	2.09E-09	1.14E-12	9.31E-12
0.1					
0.05	1.00	1.98	2.96	3.83	4.04
0.025	1.00	1.99	2.98	3.92	4.02
0.0125	1.00	1.99	2.99	3.96	4.01

Table 1. Numerical results for Example 1

The equidistant grid $\{x_{j,l}\}$ is used for realizing a second order basic discretization (DQ 2) based on symmetric finite differences according to

$$\begin{cases} F\Big(x_{1,0}, y_0, y_0', \frac{\frac{Y^0_{1,1}-y_0}{h}-y_0'}{\frac{h}{2}}\Big) = 0; \\ \text{For } j \geqslant 1,\, l > 1: \\ \;F\Big(x_{j,l}, Y^0_{j,l}, \frac{Y^0_{j,l+1}-Y^0_{j,l-1}}{2h}, \frac{Y^0_{j,l+1}-2Y^0_{j,l}+Y^0_{j,l-1}}{h^2}\Big) = 0; \\ \text{For } j > 1,\, l = 1: \\ \;F\Big(x_{j-1,m}, Y^0_{j-1,m}, \frac{Y^0_{j,1}-Y^0_{j-1,m-1}}{2h}, \\ \qquad\qquad \frac{Y^0_{j,1}-2Y^0_{j-1,m}+Y^0_{j-1,m-1}}{h^2}\Big) = 0. \end{cases}$$

The Lobatto grid $\{\hat{x}_{j,l}\}$ is used for the computation of an interpolated defect. This is realized as follows:

- First, after interpolating the current iterate and defining the defect in the usual way, the defect is evaluated at the $\hat{x}_{j,k}$,

$$\hat{d}^i_{j,k} := \tfrac{d}{dx}\,p^i(x_{j,k}) - f(x^i, p^i(x_{j,k})), \quad k = 1 \ldots \hat{m}.$$

- Next, after interpolating the $\hat{d}^i_{j,k}$ by a piecewise polynomial function $\hat{d}(x)$ we define the modified defect

$$\begin{cases} \hat{d}^{A,i}_{1,0} := \hat{d}^{\,i}(x_{1,0}), \quad \hat{d}^{B,i}_{1,0} := \tfrac{d}{dx}\,p^i(x_{1,0}) - y_0'; \\ \text{For } j \geqslant 1,\, l > 0: \quad \hat{d}^i_{j,l} := \hat{d}^{\,i}(x_{j,l}); \\ \text{For } j > 1,\, l = 0: \\ \;\hat{d}^i_{j,0} = F\Big(x_{j,l}, p^i_{j,0}, \frac{\frac{d}{dx}\,p^i(x_{j,0})+\frac{d}{dx}\,p^i(x_{j-1,m})}{2}, \hat{\gamma}^i_{j,0}\Big) \end{cases} \tag{20a}$$

with the 'jump defect'

$$\hat{\gamma}^i_{j,0} := \frac{\frac{d^2}{dx^2}\,p^i(x_{j,0}) + \frac{d^2}{dx^2}\,p^i(x_{j-1,m})}{2} + \frac{\frac{d}{dx}\,p^i(x_{j,0}) - \frac{d}{dx}\,p^i(x_{j-1,m})}{h}. \tag{20b}$$

Then we solve the corresponding discretized neighboring problem

$$\begin{cases} F\Big(x_{1,0}, y_0, y_0' + \hat{d}^{B,i}_{1,0}, \frac{\frac{\tilde{Y}^i_{1,1}-y_0}{h}-(y_0'+d^{B,i}_{1,0})}{\frac{h}{2}}\Big) = d^{A,i}_{1,0}; \\ \text{For } j \geqslant 1,\, l > 1: \\ \;F\Big(x_{j,l}, \tilde{Y}^i_{j,l}, \frac{\tilde{Y}^i_{j,l+1}-\tilde{Y}^i_{j,l-1}}{2h}, \frac{\tilde{Y}^i_{j,l+1}-2\tilde{Y}^i_{j,l}+\tilde{Y}^i_{j,l-1}}{h^2}\Big) = \hat{d}^i_{j,l}; \\ \text{For } j > 1,\, l = 1: \\ \;F\Big(x_{j-1,m}, \tilde{Y}^i_{j-1,m}, \frac{\tilde{Y}^i_{j,1}-\tilde{Y}^i_{j-1,m-1}}{2h}, \frac{\tilde{Y}^i_{j,1}-2\tilde{Y}^i_{j-1,m}+\tilde{Y}^i_{j-1,m-1}}{h^2}\Big) = \hat{d}^i_{j,0}, \end{cases}$$

and proceed as before (cf. (17)).

The purpose of the modified defect definition (20) is, like for classical explicit first order IPDeC from [3], to modify the iteration in such a way that its fixed point is given by a higher-order superconvergent collocation scheme, in our case of Lobatto type. In fact, Lobatto collocation at the nodes $\hat{x}_{j,k}$ means that the defect of the collocation polynomial vanishes at these nodes, and thus, this collocation polynomial is a fixed point of our iteration. This lets us expect that after several defect correction steps a superconvergent iterate is obtained; see Example 2 for numerical evidence.

1. Remark. In (20a), a defect with respect to the initial condition for the first derivative is also taken into account. Furthermore, the discontinuity of the first derivative of $p^i(x)$ at the endpoints of the intervals $\mathbf{I}_j$ $(p^i(\mathbf{x}_j))$ enforces to include the jump defect $\hat{\gamma}^i_{j,0}$, see (20), see also [9].

Example 2. Consider the implicit scalar nonlinear test problem

$$e^{y''(x)} + y'(x) + y(x) \tag{21a}$$
$$= e^{-\sin x} + 1 + \sin x + \cos x,$$
$$y(0) = 1, \;\; y'(0) = 1, \tag{21b}$$

with exact solution $y^*(x) = 1+\sin x$. The numerical solution is computed over a sequence of subintervals of length **h**, each of them divided into 6 equidistants nodes and 5 Lobatto nodes ($\hat{c}_1 = 0$, $\hat{c}_2 = \frac{1}{2} - \frac{\sqrt{21}}{14}$, $\hat{c}_3 = \frac{1}{2}$;, $\hat{c}_4 = \frac{1}{2} + \frac{\sqrt{21}}{14}$, $\hat{c}_5 = 1$).

We choose the integration interval $x \in [0, 3]$. The resulting global errors with respect to the exact solution at the endpoint $x = 3$ are displayed in Table 2 together with the observed convergence orders. Results are given for the basic scheme (DQ 2), 4 IIPDeC 2 iterates working in passive mode, and the fixed point of the IIPDeC 2 iteration (L-COLL, corresponding to Lobatto collocation of degree $\hat{m} = 5$ at the nodes $\hat{x}_{j,k}$ where the defect is interpolated). Note that the convergence order of the Lobatto collocation scheme is

h	DQ 2	IIPDeC 2/1	IIPDeC 2/2	IIPDeC 2/3	L-COLL
0.1	2.30E-05	1.93E-09	9.62E-14	6.07E-16	6.32E-16
0.05	5.75E-06	1.21E-10	1.51E-15	2.37E-18	2.47E-18
0.025	1.44E-06	7.54E-12	2.36E-17	9.24E-21	9.63E-21
0.0125	3.59E-07	4.71E-13	3.69E-19	3.61E-23	3.76E-23
0.1	2.00	4.00	5.99	8.00	8.00
0.05	2.00	4.00	6.00	8.00	8.00
0.025	2.00	4.00	6.00	8.00	8.00
0.0125					

TABLE 2. Numerical results for Example 2.

$\mathcal{O}(h^{2\hat{m}-2}) = \mathcal{O}(h^8)$, and the same convergence order is realized after only 3 IIPDeC 2 iteration steps.

REFERENCES

1. *Auzinger W.* Defect correction methods. — Encyclopedia of Applied and Computational Mathematics, Volume 1, Enquist, B. (Ed.), Berlin: Springer, 2015. — p. 323–332,
2. *Auzinger W.* Error estimation via defect computation and reconstruction: Some particular techniques // J. Numer. Anal. Ind. Appl. Math. — 2011. — **6**, №1-2. — P. 15–27.
3. *Auzinger W., Hofstätter H., Kreuzer W., Weinmüller E.* Modified defect correction algorithms for ODEs. Part I: General theory // Numer. Algorithms. — 2004. — **36**, №2. — P. 135–155.
4. *Auzinger W., Hofstätter H., Kreuzer W., Weinmüller E.* Modified defect correction algorithms for ODEs. Part II: Stiff initial value problems // Numer. Algorithms. — 2005. — **40**, №3. — P. 285–303.
5. *Auzinger W., Koch O., Weinmüller E.* Efficient collocation schemes for singular boundary value problems // Numer. Algorithms. — 2002. — **31**, №1. — P. 5–25.
6. *Böhmer W., Stetter H.J., Eds.* Defect Correction Methods – Theory and Applications. — Computing Suppl. 5, Berlin: Springer-Verlag, 1984.
7. *Butcher J.C., Cash J.R., Moore G., Russell R.D.* Defect correction for two-point boundary value problems on nonequidistant meshes // Math. Comput. — 1995. — **64**, №210. — P. 629–648.
8. *Dutt A., Greengard L., Rokhlin V.* Spectral deferred correction methods for ordinary differential equations // BIT Numer. Math. — 2000. — **40**, №2. — P. 241–266.
9. *Frank R.* The method of iterated defect-correction and its application to two-point boundary value problems // Numer. Math. — 1975. — **25**, №4. — P. 409–419.
10. *Frank R, Ueberhuber C.W.* Iterated defect correction for differential equations. Part I: Theoretical results // Computing. — 1978. — **20**, №3. — P. 207–228.
11. *Gavriluk I.P., Hermann M., Makarov V.L., Kutniv M.V.* Exact and Truncated Difference Schemes for Boundary Value ODEs. — Basel: Birkhäuser, 2011.
12. *Hansen A.C., Strain J.* On the order of deferred correction // Appl. Numer. Math. — 2011. — **61**, №8. — P. 961–973.
13. *Hairer A., Nørsett S.P., Wanner, G.* Solving Ordinary Differential Equations I. Nonstiff Problems. — Berlin: Springer, 1993.

14. *Pereyra V.* Iterated deferred correction for nonlinear boundary value problems // Numer. Math. — 1968. — **11**, №2. — P. 111–125.
15. *Samarskii A.A.* The Theory of Difference Schemes. — New-York, Basel: Marcel Dekker Inc., 2001.
16. *Schild, K.H.* Gaussian collocation via defect correction // Numer. Math. — 1990. — **58**, №1. — P. 369–386.
17. *Skeel R.D.* A theoretical framework for proving accuracy results for deferred corrections // SIAM J. Numer. Anal. — 1981. — **19**, №1. — P. 171–196.
18. *Stetter H.J.* The defect correction principle and discretization methods // Numer. Math. — 1978. — **29**, №4. — P. 425–443.
19. *Tutz M.* Iterierte Defektkorrektur für explizite und implizite Anfangswertprobleme erster und zweiter Ordnung: PhD Thesis (in German). — Vienna University of Technology, 2013.
20. *Zadunaisky P.E.* On the estimation of errors propagated in the numerical integration of ODEs // Numer. Math. — 1976. — **27**, №1. — P. 21–39.

Стаття: надійшла до редколегії 15.05.2016
прийнята до друку 20.02.2017

МЕТОДИ КОРЕКЦІЇ ДЕФЕКТУ, КЛАСИЧНІ ТА НОВІ

Вінфрід АУЦІНҐЕР[1], Роксолана СТОЛЯРЧУК[2], Мартін ТУТЦ[1]

[1]*Віденьський технічний університет,*
Віднер Гауптштрассе, 8-10, 1040 Відень, Австрія
[2]*Національний університет "Львівська Політехніка",*
вул. С. Бандери 12, 79013, Львів, Україна

Методи корекції дефекту ґрунтуються на ідеї оцінки точності наближеного розв'зку за допомогою формування дефекту, або залишку, стосовно до даної задачі. За допомогою процедури зворотнього розв'язування отримуємо оцінку похибки. Цей процес можна продовжити ітеративно. Мета цього огляду – подальше поширення концепції, що розглядається. Більш того, вперше подано загальний і узгоджений огляд різних типів методів корекції дефекту, їхнє застосувань в контексті дискретизаційних схем для диференціальних рівнянь. Після опису загального алгоритму обговоримо деякі спеціальні технології, які використовуються для розв'язування звичайних диференціальних рівнянь. Також представлені нові результати стосовно застосування до неявних задач.

Ключові слова: корекція дефекту, дискретизація, звичайні диференціальні рівняння.