

An Extension of the Applicability of Iterated Deferred Corrections

By Reinhard Frank, Joerg Hertling and Christoph W. Ueberhuber

Abstract. A new way of estimating local discretization errors (based on an idea due to P. E. Zadunaisky) is introduced. If error estimates obtained by this method are used in connection with the general class of iterated deferred correction algorithms, they lead to an extension of the domain of applicability, when compared with the variants used by Fox and Pereyra.

1. Introduction. Iterated Deferred Correction (IDC) has been introduced by Fox [2] and Pereyra (e.g. [10]–[13]) as a technique to accelerate the convergence of finite difference schemes. This procedure is based on estimating the local discretization error, and all implementations known to the authors (e.g. Pereyra [13], Lentini, Pereyra [8], Daniel, Martin [1]) require an *explicit* knowledge of the asymptotic expansion of the local discretization error. This severely restricts the range of applications of iterated deferred corrections; if the structure of the required asymptotic expansions is too complicated, an automatic deferred correction solver cannot be implemented in a practicable manner.

In this paper we will present a new technique for obtaining estimates of the local truncation error, which is based on an idea of P. E. Zadunaisky [15]. This procedure does *not* require the explicit knowledge of the corresponding asymptotic expansions. So, even extremely involved expansions of the local discretization error do not cause any difficulties for an automatic IDC-code which utilizes this estimation procedure. In Sections 2 and 3 we will develop and discuss in detail our variants of the IDC. Section 4 contains an asymptotic analysis ($h \rightarrow 0$), and the results of numerical experiments are presented in Section 5.

2. Basic Ideas. To outline our ideas we will use scalar two-point boundary-value problems (BVPs) of the following form:

$$(2.1) \quad y'' = f(t, y, y'), \quad y(0) = A, \quad y(1) = B$$

with f sufficiently smooth. This type of problem has been chosen because all “classical” implementations of deferred corrections are not immediately applicable to (2.1) if f depends nonlinearly on y' (cf. Pereyra [12]). To apply standard IDC-codes it would be necessary to transform (2.1) into a first-order system. Such a transformation has the practical disadvantage of doubling the dimension of the system of nonlinear algebraic equations which has to be solved at each step of the IDC. Using the new variant of the IDC presented below, (2.1) may be solved without transforming it into a first-order system.

Received June 14, 1976; revised November 18, 1976.

AMS (MOS) subject classifications (1970). Primary 34A50, 65L10, 65B05.

Copyright © 1977, American Mathematical Society

Our ideas may be applied to rather arbitrary types of operator equations; we will use the two-point BVP (2.1) to explain the ideas and to analyze its asymptotic behavior:

Consider the “classical” second-order finite difference scheme:

$$(2.2) \quad (\eta_{\nu-1}^0 - 2\eta_{\nu}^0 + \eta_{\nu+1}^0)/h^2 = f(t_{\nu}, \eta_{\nu}^0, (\eta_{\nu+1}^0 - \eta_{\nu-1}^0)/2h), \quad \nu = 1(1)n-1, \\ \eta_0^0 = A, \quad \eta_n^0 = B$$

with the solution-vector $\eta^0 := (\eta_0^0, \dots, \eta_n^0)$ which is an approximation to the exact solution y of (2.1) at the (equidistant) gridpoints $t_{\nu} = \nu \cdot h, \nu = 0(1)n; h = 1/n$. If we knew the *exact* local discretization error

$$(2.3) \quad l_{\nu} := (y(t_{\nu-1}) - 2y(t_{\nu}) + y(t_{\nu+1}))/h^2 \\ - f(t_{\nu}, y(t_{\nu}), (y(t_{\nu+1}) - y(t_{\nu-1}))/2h),$$

at every gridpoint t_{ν} we could obtain the vector $\eta = (\eta_0, \dots, \eta_{\nu}, \dots, \eta_n) = (y(0), \dots, y(t_{\nu}), \dots, y(1))$ of the discretized exact solution $y(t)$ by solving the system

$$(2.4) \quad (\eta_{\nu-1} - 2\eta_{\nu} + \eta_{\nu+1})/h^2 = f(t_{\nu}, \eta_{\nu}, (\eta_{\nu+1} - \eta_{\nu-1})/2h) + l_{\nu}, \quad \nu = 1(1)n-1, \\ \eta_0 = A, \quad \eta_n = B.$$

The well-known idea of IDC consists in using η^0 in order to obtain an estimate $l^1 := (l_1^1, \dots, l_{n-1}^1)$ of the local discretization error $l := (l_1, \dots, l_{n-1})$ and in solving

$$(2.5) \quad (\eta_{\nu-1}^1 - 2\eta_{\nu}^1 + \eta_{\nu+1}^1)/h^2 = f(t_{\nu}, \eta_{\nu}^1, (\eta_{\nu+1}^1 - \eta_{\nu-1}^1)/2h) + l_{\nu}^1, \quad \nu = 1(1)n-1, \\ \eta_0^1 = A, \quad \eta_n^1 = B$$

to obtain the improved numerical solution $\eta^1 := (\eta_0^1, \dots, \eta_n^1)$. This idea may be used in an iterative fashion: The estimates of the local truncation error $l^2, l^3, \dots, l^j, \dots$ which are needed to compute the approximate solutions $\eta^2, \eta^3, \dots, \eta^j, \dots$ are obtained from $\eta^1, \eta^2, \dots, \eta^{j-1}, \dots$, respectively.

We will now describe our method for obtaining the quantities l_{ν}^j :

Consider a fixed gridpoint t_{ν} , and let $t_{\mu}, \dots, t_{\nu-1}, t_{\nu+1}, \dots, t_{\mu+m_j}$ be m_j adjacent gridpoints. Define the polynomial P_{ν}^j of degree m_j which interpolates to η^{j-1} at the points $t_{\mu}, \dots, t_{\mu+m_j}$:

$$(2.6) \quad P_{\nu}^j(t_{\kappa}) = \eta_{\kappa}^{j-1}, \quad \kappa = \mu(1)\mu + m_j.$$

We use P_{ν}^j in order to define the following *new BVP*

$$(2.7) \quad y'' = f(t, y, y') + (P_{\nu}^j)''(t) - f(t, P_{\nu}^j(t), (P_{\nu}^j)'(t)), \\ y(0) = P_{\nu}^j(0), \quad y(1) = P_{\nu}^j(1)$$

whose *exact* solution is P_{ν}^j . Since the values η_{κ}^{j-1} are (assumed to be) good approximations to $y(t_{\kappa})$, we may expect that $P_{\nu}^j(t)$, $(P_{\nu}^j)'(t)$ and $(P_{\nu}^j)''(t)$ approximate $y(t)$, $y'(t)$ and $y''(t)$, respectively, on $[t_{\mu}, t_{\mu+m_j}]$. Outside of $[t_{\mu}, t_{\mu+m_j}]$ the polynomial $P_{\nu}^j(t)$ may differ significantly from $y(t)$. Therefore, we could think of (2.7) as a “local neighboring problem” for (2.1). Since we know the *exact* solution of the BVP (2.7) we have the *exact* local discretization error l_{ν}^j of (2.7) at the gridpoint t_{ν} at our

disposal. The “local proximity” of (2.1) and (2.7) near t_ν suggests to use the *known* local discretization error l_ν^j as an estimate for the *unknown* discretization error l_ν .

The idea of constructing a “neighboring problem” with known solution (using the results of a numerical method applied in advance) and to obtain error estimates via the numerical solution of this “neighboring problem” is due to Zadunaisky [15]. He used this idea to estimate the *global* discretization error of Runge-Kutta methods. The proposal to use this idea iteratively may be found in Stetter [14]. An asymptotic analysis ($h \rightarrow 0$) of such an iterative procedure (based on global error estimates) has been given by Frank [4] and Frank, Ueberhuber [6]. In these papers the method has been called Iterated *Defect* Correction (IDeC). According to this course of development, the above scheme (i.e. our special variant of the IDC) may be considered as a “local variant” of the IDeC.

3. Algorithmic Details. We now describe some practical details of the procedure introduced in Section 2. An essential point of our method is the construction of “local neighboring problems” (cf. (2.7)) with known solution.

Since the local discretization error of such a problem, at the point t_ν , is

$$(3.1) \quad \begin{aligned} l_\nu^j &= [P_\nu^j(t_{\nu-1}) - 2P_\nu^j(t_\nu) + P_\nu^j(t_{\nu+1})]/h^2 \\ &\quad - [f(t_\nu, P_\nu^j(t_\nu), (P_\nu^j(t_{\nu+1}) - P_\nu^j(t_{\nu-1}))/2h) \\ &\quad + (P_\nu^j)''(t_\nu) - f(t_\nu, P_\nu^j(t_\nu), (P_\nu^j)'(t_\nu))], \end{aligned}$$

only the values $(P_\nu^j)'(t_\nu)$ and $(P_\nu^j)''(t_\nu)$ are needed for an implementation [the required values of P_ν^j are given by $P_\nu^j(t_{\nu-1}) = \eta_{\nu-1}^{j-1}$, $P_\nu^j(t_\nu) = \eta_\nu^{j-1}$, $P_\nu^j(t_{\nu+1}) = \eta_{\nu+1}^{j-1}$]. Therefore, it is not necessary to set up the polynomials P_ν^j explicitly. It suffices to calculate the quantities $(P_\nu^j)'(t_\nu)$ and $(P_\nu^j)''(t_\nu)$ by forming weighted sums

$$(3.2) \quad \begin{aligned} (P_\nu^j)'(t_\nu) &= (1/h)[W_0^{v-\mu}\eta_\mu^{j-1} + \dots + W_{m_j}^{v-\mu}\eta_{\mu+m_j}^{j-1}], \\ (P_\nu^j)''(t_\nu) &= (1/h^2)[V_0^{v-\mu}\eta_\mu^{j-1} + \dots + V_{m_j}^{v-\mu}\eta_{\mu+m_j}^{j-1}], \end{aligned}$$

where the superscript $v - \mu$ of the weights denotes their dependence on the relative position of t_ν with respect to the interval $[t_\mu, t_{\mu+m_j}]$ (cf. (2.6)).

Of the many different ways to choose $[t_\mu, t_{\mu+m_j}]$ around t_ν we mention only two possibilities:

(a) For even values of $m_j (= 2r_j)$ we might choose $[t_\mu, t_{\mu+m_j}]$ *symmetrically* around t_ν : $[t_{\nu-r_j}, t_{\nu+r_j}]$. In order to use symmetric intervals near the boundaries the well-known trick devised by Fox (e.g. Fox [2, pp. 69–71], Keller [7]), may be applied. The idea is simply to compute auxiliary values $\eta_{-1}^{j-1}, \eta_{-2}^{j-1}, \dots, \eta_{n+1}^{j-1}, \eta_{n+2}^{j-1}, \dots$ at points exterior to $[0, 1]$, provided that we do not approach a possible singularity too closely. To use the symmetric intervals $[t_{\nu-r_j}, t_{\nu+r_j}]$ around t_ν means that there is exactly one local neighboring problem (2.7) for every gridpoint t_ν , i.e. $[t_{\nu-r_j}, t_{\nu+r_j}]$ is moving along $[0, 1]$.

(b) If n is a multiple of m_j ($n = s_j m_j$) for all IDC-steps j , the interval $[0, 1]$ is subdivided into intervals $[0, t_{m_j}], [t_{m_j}, t_{2m_j}], \dots, [t_{(s_j-1)m_j}, 1]$. In this case only *one* local neighboring problem is defined on every subinterval. For $t_\nu \in (t_{(i-1)m_j}, t_{im_j})$

the respective problem (2.7) is defined by means of the polynomial P_i^j , which interpolates $\eta_{(i-1)m_j}^{j-1}, \dots, \eta_{im_j}^{j-1}$. In contrast to (a) no trick has to be applied near the boundaries $t_0 = 0$ and $t_n = 1$.

Concerning the differentiation weights, one preferably uses variant (a) since only *one* set of weights (i.e. $W_\kappa^{\nu-\mu} = W_\kappa^{\nu-(\nu-r_j)} = W_\kappa^r j$, and $V_\kappa^r j$, $\kappa = 0(1)m_j$) is needed.

In contrast to the IDeC based on global error estimates (outlined in Frank [4]), where a constant degree of the interpolating polynomials (for the different steps of the IDeC) is an essential feature of the method, it is advisable in our variant of the IDC to use increasing degrees of the interpolating polynomials: $m_1 < m_2 < m_3 < \dots$. For example, a reasonable choice would be $m_j = 2j + 2$. Further details of the implementation of IDC-schemes (e.g. mesh selection algorithms, stopping criteria, ...) will *not* be discussed here, but we refer the reader to Lentini, Pereyra [8] and Daniel, Martin [1].

4. Asymptotic Behavior. An extensive asymptotic analysis of the IDC-methods applied to operator equations has been given by Pereyra [11]. Since our variant is a member of the general class of IDC-methods considered there, we can use the results of Pereyra (cf. Theorem 3.2 and Theorem 3.3 of Pereyra [11]).

We restrict our considerations to problem (2.1) and assume an IDC-method as described in Section 3: We consider interpolating polynomials with increasing degree $m_j = 2j + 2$ located symmetrically to t_ν (version (a) of Section 3). To prove the existence of asymptotic expansions it is essential to use a fixed formula over the whole interval $[0, 1]$. Therefore, we use the extension technique of Fox near the boundaries, i.e. values η_ν^j are computed on intervals I_h^j with $I_h^0 \supset I_h^1 \supset \dots \supset I_h^{j_{\max}} = [0, 1]$, where the size of these intervals is determined by the stepsize h and the intended maximum number of iterations j_{\max} . As the following theorem is concerned with asymptotic results ($h \rightarrow 0$) we will choose the intervals (for theoretical reasons only) as being independent of h . If h_0 denotes the largest stepsize considered, we choose $I^j := I_{h_0}^j \supset I_h^j$. As a result of this assumption, we use redundant values η_ν^j at gridpoints t_ν with $t_\nu \in I^j$ but $t_\nu \notin I_h^j$, which are irrelevant for practical implementations.

Theorem 3.2 of Pereyra [11] reads now:

THEOREM 4.1. *From the inductive assumption*

$$(4.1) \quad \eta_\nu^{j-1} - y(t_\nu) = h^{2j} e_{2j}^{j-1}(t_\nu) + h^{2j+2} e_{2j+2}^{j-1}(t_\nu) + \dots + h^{2J} e_{2J}^{j-1}(t_\nu) + R_\nu$$

with $R_\nu = O(h^{2J+2})$ on I^{j-1} , it follows that

$$(4.2) \quad \eta_\nu^j - y(t_\nu) = h^{2j+2} e_{2j+2}^j(t_\nu) + h^{2j+4} e_{2j+4}^j(t_\nu) + \dots + h^{2J-2} e_{2J-2}^j(t_\nu) + \bar{R}_\nu$$

with $\bar{R}_\nu = O(h^{2J})$ on I^j .

Proof. The assertion (4.2) is identical with (3.9) of Theorem 3.2 of Pereyra [11]. Therefore, we only have to show that the hypotheses of this theorem hold for our special case, as in particular we need to verify the assumption (3.4) of Lemma 3.1 (Pereyra [11]).

Our error-estimation procedure determines completely the operators S_j introduced by Pereyra (cf. (3.1) of Pereyra [11]). The ν th component $[S_j(\eta)]_\nu$ of S_j for a given argument $\eta = (\eta_{-\rho}, \dots, \eta_0, \dots, \eta_\nu, \dots, \eta_n, \dots, \eta_{n+\rho})$ is defined as

$$\begin{aligned}
 [S_j(\eta)]_\nu &:= [P_\nu(t_{\nu-1}) - 2P_\nu(t_\nu) + P_\nu(t_{\nu+1})]/h^2 \\
 &\quad - f(t_\nu, P_\nu(t_\nu), [P_\nu(t_{\nu+1}) - P_\nu(t_{\nu-1})]/2h) \\
 &\quad - P_\nu''(t_\nu) + f(t_\nu, P_\nu(t_\nu), P_\nu'(t_\nu)),
 \end{aligned}
 \tag{4.3}$$

where P_ν is the polynomial (of degree $2j + 2$) defined by

$$P_\nu(t_\kappa) = \eta_\kappa, \quad \kappa = \nu - j - 1(1)\nu + j + 1.$$

Note. The components $\eta_{-\rho}, \dots, \eta_{-1}; \eta_{n+1}, \dots, \eta_{n+\rho}$ correspond to the gridpoints outside of $[0, 1]$, which are necessary to enable a uniform definition of $[S_j(\eta)]_\nu$ for all relevant values of ν according to Fox's device (cf. Section 3 and the remarks preceding Theorem 4.1). ρ decreases with increasing j ; and therefore, the dimension of the vector $S_j(\eta)$ is smaller than the dimension of the vector η .

In (3.4) of [11] the operators S_j are applied to the vectors $\eta = \varphi_h y := (y(-\rho h), \dots, y(0), \dots, y(t_\nu), \dots, y(1), \dots, y(1 + \rho h))$, where $y(t)$ is the solution of problem (2.1). The polynomials P_ν^* defined by

$$P_\nu^*(t_\kappa) = y(t_\kappa), \quad \kappa = \nu - j - 1(1)\nu + j + 1,$$

satisfy

$$\begin{aligned}
 &[(d^i/dt^i)P_\nu^*](t_\nu) \\
 &= [(d^i/dt^i)y](t_\nu) + \text{const } h^{2s} [(d^r/dt^r)y](t_\nu) \\
 &\quad + \text{const } h^{2s+2} [(d^{r+2}/dt^{r+2})y](t_\nu) + \dots + O(h^{2J}),
 \end{aligned}
 \tag{4.4}$$

where

$$\begin{aligned}
 s &:= (j + 2) - i/2, & r &:= 2j + 4 \quad \text{if } i \text{ is even,} \\
 s &:= (j + 2) - (i + 1)/2, & r &:= 2j + 3 \quad \text{if } i \text{ is odd.}
 \end{aligned}
 \tag{4.5}$$

$[S_j(\varphi_h y)]_\nu$ may be expanded as

$$\begin{aligned}
 [S_j(\varphi_h y)]_\nu &= [P_\nu^*(t_{\nu-1}) - 2P_\nu^*(t_\nu) + P_\nu^*(t_{\nu+1})]/h^2 \\
 &\quad - f(t_\nu, P_\nu^*(t_\nu), [P_\nu^*(t_{\nu+1}) - P_\nu^*(t_{\nu-1})]/2h) \\
 &\quad - (P_\nu^*)''(t_\nu) + f(t_\nu, P_\nu^*(t_\nu), (P_\nu^*)'(t_\nu)) \\
 &= (2h^2/4!)(P_\nu^*)^{\text{IV}}(t_\nu) + (2h^4/6!)(P_\nu^*)^{\text{VI}}(t_\nu) \\
 &\quad + \dots - f(t_\nu, P_\nu^*(t_\nu), (P_\nu^*)'(t_\nu) + (h^2/3!)(P_\nu^*)'''(t_\nu) + \dots) \\
 &\quad + f(t_\nu, P_\nu^*(t_\nu), (P_\nu^*)'(t_\nu)) \\
 &= (2h^2/4!)(P_\nu^*)^{\text{IV}}(t_\nu) + (2h^4/6!)(P_\nu^*)^{\text{VI}}(t_\nu) \\
 &\quad + \dots - \text{const } h^2 f_{y'}(t_\nu, P_\nu^*(t_\nu), (P_\nu^*)'(t_\nu))(P_\nu^*)'''(t_\nu) \\
 &\quad - h^4 [\text{const } f_{y'y'}(t_\nu, P_\nu^*(t_\nu), (P_\nu^*)'(t_\nu))(P_\nu^*)^{\text{V}}(t_\nu) \\
 &\quad \quad + \text{const } f_{y'y'y'}(t_\nu, P_\nu^*(t_\nu), (P_\nu^*)'(t_\nu))(P_\nu^*)'''(t_\nu)^2] \\
 &\quad - h^6 [\dots] - \dots + O(h^{2J}).
 \end{aligned}
 \tag{4.6}$$

Using $P_\nu^*(t_\nu) = y(t_\nu)$, substituting (4.4) into (4.6), applying further Taylor expansions and reordering the terms of the resulting expression according to powers of h , we explicitly obtain the operators \bar{F}_j and $t_{j\lambda}$ of (3.4) in Pereyra [11]. After forming Fréchet-derivatives, the desired relation (3.4) of Pereyra [11] is verified. \square

Note. An alternative proof of this theorem is given in Frank, Hertling, Ueberhuber [5] for problems of the type $y'' = f(t, y)$. The generalization to problems (2.1) is straightforward.

5. Numerical Results. The technique described above has been applied to a number of test problems. Here we only show the application to the following simple test problem (equation of the *catenary*)

$$(5.1) \quad \begin{aligned} y'' &= \sqrt{1 + (y')^2}, \\ y(-1) &= \cosh(-1), \\ y(1) &= \cosh(1), \end{aligned}$$

with the solution $y = \cosh(t)$. The error estimates at the boundaries were obtained by using auxiliary points outside of $[-1, 1]$ (cf. Section 3); the degrees of the polynomials P_ν^j were $m_j = 2j + 2$. Stepsizes $h = 1/4$ and $h = 1/8$ have been used. The results

TABLE 1
Absolute errors for problem (5.1)

		h = 1/4	h = 1/8
j = 0	min.	9.80 -4	1.27 -4
	mean	1.75 -3	4.11 -4
	max.	2.38 -3	5.92 -4
j = 1	min.	1.43 -5	4.55 -7
	mean	2.53 -5	1.46 -6
	max.	3.42 -5	2.10 -6
j = 2	min.	2.03 -7	1.67 -9
	mean	3.19 -7	4.63 -9
	max.	3.96 -7	6.15 -9
j = 3	min.	2.06 -10	1.66 -12
	mean	2.54 -9	9.38 -12
	max.	3.89 -9	1.48 -11
j = 4	min.	3.82 -11	6.39 -14
	mean	2.01 -10	1.73 -13
	max.	3.39 -10	3.06 -13
j = 5	min.	1.55 -11	9.95 -14
	mean	3.55 -11	3.67 -13
	max.	6.25 -11	5.33 -13

TABLE 2
*Absolute errors and orders of convergence for
 problem (5.1) at the gridpoint $t = 0$. (The
 symbol * denotes the fact that machine
 accuracy has been reached).*

	$h = 1/4$	$h = 1/8$
$j = 0$		
abs.error	2.38 -3	5.92 -4
order		2.01
$j = 1$		
abs.error	3.42 -5	2.10 -6
order		4.03
$j = 2$		
abs.error	3.96 -7	6.15 -9
order		6.01
$j = 3$		
abs.error	2.06 -10	1.66 -12
order		6.96
$j = 4$		
abs.error	3.39 -10	1.28 -13*
order		11.37
$j = 5$		
abs.error	5.04 -11	5.33 -13*
order		6.56

of this section were obtained with a FORTRAN implementation on the CDC Cyber 74 at the Technical University of Vienna in single precision (48-bit mantissa).

In Table 1 we present a summary of the absolute errors obtained on an equidistant grid. Table 2 contains the absolute errors at the point $t = 0$ for both step-sizes and the numerically calculated orders. Compare these orders with the orders 2, 4, 6, 8, 10, 12 which are theoretically to be expected.

Results obtained from other test problems, supported our theoretical considerations in a similar way.

6. Conclusion. A new variant of the well-known Iterated Deferred Correction (IDC) methods has been presented in this paper. In contrast to all IDC-methods implemented so far, this new variant is not based on the explicit knowledge of the asymptotic expansion of the local discretization error. Consequently, it is suitable for operator equations where this requirement presents difficulties to conventional imple-

mentations of IDC-methods. But problems of that type could also be solved with Iterated *Defect* Correction (IDeC) methods (e.g. Frank [4], Frank, Ueberhuber [6]). For example, BVPs of the type $y'' = f(t, y, y')$ have been solved successfully with both methods: the IDeC and the new variant of the IDC, discussed in this paper.

The question arises then of when should one prefer one algorithm or the other.

The advantage of all IDC-methods (compared with IDeC-methods) lies in the possibility to increase the order of convergence of the scheme without limitations (at least theoretically) in the course of the procedure, whereas the IDeC-methods assume an a priori choice of the maximum attainable order of convergence (because polynomials of a fixed degree have to be used). A second advantage of deferred corrections is that it preserves the structure of the Jacobians (e.g. tridiagonal matrices for problem (2.1)). If the IDeC is applied to second-order problems, difficulties arise at those points where different interpolating polynomials join. This fact may sometimes lead to a perturbed structure of the Jacobian (e.g. for problem (2.1) the tridiagonal form of the Jacobian is slightly perturbed by elements which increase the bandwidth).

From these remarks no conclusive decision may be drawn on which method is preferable for a given problem. Insights, necessary for such a decision, could only be gained in a large-scale test-study which should also include the new method of B. Lindberg which has recently been brought to our attention (cf. Lindberg [9]). This method may also be considered as a variant of IDC.

Acknowledgement. The authors wish to thank the referee for his valuable comments on the first version of this paper.

Institut für Numerische Mathematik
Technische Universität Wien
Wien, Austria

1. J. W. DANIEL & A. J. MARTIN, *Implementing Deferred Corrections for Numerov's Difference Method for Second-Order Two-Point Boundary-Value Problems*, Report CNA-107, Center for Numerical Analysis, Univ. of Texas, Austin, Texas, 1975.
2. L. FOX, *The Numerical Solution of Two-Point Boundary Value Problems in Ordinary Differential Equations*, Oxford Univ. Press, New York, 1957. MR 21 #972.
3. L. FOX (Editor), *Numerical Solution of Ordinary and Partial Differential Equations*, Pergamon Press, Oxford, 1962, pp. 38–39. MR 26 #4488.
4. R. FRANK, "The method of iterated defect-correction and its application to two-point boundary value problems," Part I, *Numer. Math.*, v. 25, 1976, pp. 409–419; *ibid.*, Part II, *Numer. Math.*, v. 27, 1977, pp. 407–420.
5. R. FRANK, J. HERTLING & C. W. UEBERHUBER, *Iterated Defect Correction Based on Estimates of the Local Discretization Error*, Report No. 18/76, Inst. für Numer. Math., Tech. Univ., Vienna, 1976.
6. R. FRANK & C. W. UEBERHUBER, *Iterated Defect Correction for Runge-Kutta Methods*, Report No. 14/75, Inst. für Numer. Math., Tech. Univ., Vienna, 1975.
7. H. B. KELLER, *Numerical Solution of Two-Point Boundary-Value Problems*, Conference Board of the Math. Sci. Regional Conf. Ser. in Appl. Math., no 24, SIAM, Philadelphia, Pa., 1976, p. 34.
8. M. LENTINI & V. PEREYRA, "Boundary problem solvers for first order systems based on deferred corrections," A. K. AZIZ (Editor), *Numerical Solutions of Boundary Value Problems for Ordinary Differential Equations*, Academic Press, New York, 1975.
9. B. LINDBERG, *Error Estimation and Iterative Improvement for the Numerical Solution of Operator Equations*, Report UIUCDCS-R-76-820, Dept. of Comput. Sci., Univ. of Illinois at Urbana-Champaign, Urbana, Illinois, July, 1976.

10. V. L. PEREYRA, "On improving an approximate solution of a functional equation by deferred corrections," *Numer. Math.*, v. 8, 1966, pp. 376–391. MR 34 #3814.
11. V. L. PEREYRA, "Iterated deferred corrections for nonlinear operator equations," *Numer. Math.*, v. 10, 1967, pp. 316–323. MR 36 #4812.
12. V. L. PEREYRA, "Iterated deferred corrections for nonlinear boundary value problems," *Numer. Math.*, v. 11, 1968, pp. 111–125. MR 37 #1091.
13. V. L. PEREYRA, *High Order Finite Difference Solution of Differential Equations*, Report STAN-CS-73-348, Comput. Sci. Dept., Stanford University, Stanford, California, 1973.
14. H. J. STETTER, "Economical global error estimation," R. A. WILLOUGHBY (Editor), *Stiff Differential Systems* (Proc. Internat. Sympos., Wildbad, Germany, 1973), Plenum Press, New York and London, 1974, pp. 245–258. MR 53 #9655.
15. P. E. ZADUNAISKY, "A method for the estimation of errors propagated in the numerical solution of a system of ordinary differential equations," *Proc. Internat. Astronom. Union Symposium No. 25* (Thessaloniki, 1964), Academic Press, New York, 1966.