

SMOOTHED PREDICTOR-CORRECTOR METHODS FOR SOLVING PARTIAL
DIFFERENTIAL EQUATIONS

P. J. van der Houwen & B. P. Sommeijer

Abstract. Special predictor-corrector methods employing residue smoothing for solving semidiscrete partial differential equations are analysed. By the technique of residue smoothing the stability condition is relaxed to such an extent that the (explicit) PC methods can be applied with time steps prescribed by accuracy considerations rather than by stability considerations. The additional computational effort involved by the explicit smoothing technique used here is rather low when compared with its stabilizing effect. However, the overall accuracy may be decreased. This paper investigates the effect of residue smoothing on the accuracy.

1. INTRODUCTION

We consider the numerical solution of the initial-value problem for systems of (nonlinear) ordinary differential equations (ODEs) of the form

$$\frac{d^\nu y(t)}{dt^\nu} = f(t, y(t)), \quad \nu = 1, 2, \quad (1)$$

which arise when time-dependent partial differential equations (PDEs) are semidiscretized in space. We shall assume that the Jacobian matrix $\partial f/\partial y$ has eigenvalues located in a negative interval $[-R, 0)$. In dealing with systems of ODEs, we have to take into account that the spectral radius R is usually extremely large. Therefore, we need an integration method with a large real stability boundary. Restricting our considerations to the class of linear multistep methods, we are led to implicit methods and as a con-

sequence we are faced with the problem of solving in each integration step the implicit equation

$$y - b_0 r^\nu f(t_{n+1}, y) = \Sigma_n, \quad \Sigma_n := \sum_{i=1}^k [-a_i y_{n+1-i} + b_i r^\nu f(t_{n+1-i}, y_{n+1-i})], \quad (2)$$

where r denotes the integration step, y_{n+1-i} presents an approximation to the exact solution $y(t)$ at $t = t_{n+1-i}$ and the coefficients a_i and b_i define the linear multistep method. The solution of this equation provides a numerical approximation to $y(t)$ at $t = t_{n+1}$. In practice, equation (2) is only approximately solved and this approximate solution is accepted as the numerical approximation y_{n+1} to $y(t)$ at $t = t_{n+1}$.

In this paper, we pursue our earlier investigation of the smoothed generalized predictor-corrector (SGPC) methods for finding approximate solutions to equation (2). These methods were proposed in [2] for the first-order case $\nu = 1$, and we here we will extend them to the case $\nu = 2$. Furthermore, we analyse the effect of the relaxation parameters occurring in the SGPC scheme on the accuracy and the stability of the SGPC method for more general problems than the model problem considered in [2].

2. SGPC METHODS

We consider SGPC methods of the form

$$y^{(j)} = y^{(j-1)} - r^{(j)} S [y^{(j-1)} - b_0 r^\nu f(t_{n+1}, y^{(j-1)}) - \Sigma_n], \quad j = 1, 2, \dots, m, \quad (3)$$

where the $r^{(j)}$ are relaxation parameters, S is a smoothing matrix, and $y^{(0)}$ is an initial approximation which will be assumed to be obtained by an (explicit) linear \bar{k} -step method. Evidently, if this method converges for $m \rightarrow \infty$, then it will converge to the solution of (2). Notice that the conventional PC method is obtained if we set $r^{(j)} S = I$. Following the terminology used in PC methods, we shall call (3) an SGPC method in $P(\text{ESC})^m E$ mode.

The method (3) may be considered as a two-level iteration scheme for approximating the solution of equation (2). In [2] the more general multi-level SGPC methods have been considered, but, for the sake of transparency, we shall confine our considerations to the two-level version (3). However, from an implementational point of view, the two-level version is sometimes less attractive, so that in our numerical experiments multi-level versions of (3) are used (see Section 7.1).

As we shall explain below, the matrix S will be chosen such that applying S to a given vector $v = (v_i)$ has the effect that large differences in successive components of this vector are reduced so that the resulting components vary smoothly as a function of i . We assume that the matrix S is normalized in the sense that its eigenvalues do not exceed 1 in magnitude.

In order to see the effect of smoothing on the residue in (3), we consider the linearized equation for the SGPC error

$$e^{(j)} = e^{(j-1)} \cdot r^{(j)} S [I - b_0 Z] e^{(j-1)}, \quad j = 1, 2, \dots, m; \quad (3')$$

$$e^{(j)} := y^{(j)} - \eta, \quad Z := r^\nu \frac{\partial f(t_{n+1}, \eta)}{\partial y},$$

where η denotes the exact solution of (2). Suppose that $e^{(j)}$ is expanded in a discrete Fourier series. Since $\partial f / \partial y$ is a discrete differential operator, it will amplify high frequencies and the matrix $I - b_0 Z$ will therefore amplify the high frequency modes occurring in $e^{(j)}$, unless r is extremely small, i.e., unless r is of magnitude $1/R$ (recall that $-R$ denotes the largest negative eigenvalue of $\partial f / \partial y$ which is extremely large in magnitude in the case of semidiscrete partial differential equations). Since we want to use integration steps of realistic size, the iteration process will cause severe instabilities unless the matrix $r^{(j)} S$ is such that the high frequencies in $[I - b_0 Z] e^{(j)}$ are damped. Moreover, assuming that Z has negative eigenvalues, we will require that $b_0 > 0$ and that $r^{(j)} S$ has nonnegative eigenvalues.

From the above discussion it is clear that the conventional PC method (where $r^{(j)} S = I$) is not appropriate for approximating the solution of equation (2). In this paper, we derive various families of SGPC methods by more suitable choices of the relaxation parameters. We distinguish :

SCPC methods : Smoothed Conventional PC methods which arise by choosing $r^{(j)} S = S$.

SSPC methods : Smoothed Stabilized PC methods which arise by choosing the relaxation parameters such that the real stability boundary is more or less optimal.

$$\frac{T_{k+1}(1+2z)-1}{2(k+1)^2 z}. \quad (5)$$

This polynomial is of degree k and satisfies the above requirements. In the interval $[-1,0]$, it is bounded by 0 and $\min(1, -1/[(k+1)^2 z])$. We shall define

$$S := \frac{T_{2^q(I+2D)}-I}{2^{2q+1}D}. \quad (6)$$

By virtue of our choice $k+1 = 2^q$ and certain factorization properties of Chebyshev polynomials, this matrix S allows an efficient implementation on a computer. It can be shown that

$$S = F_1 \cdot F_2 \cdot \dots \cdot F_q,$$

where the factor matrices F_i are generated according to the recursion $F_0 = I$, $F_{i+1} = [I - 2F_i]^2$. Thus, the smoothing operator defined by (6) is a polynomial operator of degree $2^q - 1$ in D and its application to some vector v requires only q matrix-vector multiplications by the factor matrices F_i . We shall call q the *degree of smoothing*. The beauty of this factorization lies in the fact that the matrices F_i are of a simple structure so that the application of the smoothing matrix S is relatively cheap. Moreover, the actual implementation of this smoothing procedure requires only a few FORTRAN lines [2].

4. ACCURACY AND STABILITY

4.1 ACCURACY

Substitution of the smoothing matrix (6) into the error equation (3') yields

$$e^{(j)} = \alpha_j(D, Z) e^{(j-1)} = P^{(j)}(D, Z) e^{(0)}, \quad j = 1, \dots, m, \quad (7)$$

where the amplification matrices $\alpha_j(D, Z)$ and $P^{(j)}(D, Z)$ are polynomials in D and Z defined by

$$P^{(j)}(D, Z) = \prod_{i=1}^j \alpha_i(D, Z), \quad j = 1, \dots, m,$$

$$\alpha_j(D, Z) := I - r^{(j)} S [I - b_0 Z] = I - r^{(j)} \frac{T_{2^q(I+2D)}-I}{2^{2q+1}D} [I - b_0 Z]. \quad (8)$$

In the reduction of the low frequencies in the SGPC error, the first few Taylor terms of the amplification factors $\alpha_j(D, Z)$ play a central role. It is easily verified that

$$\alpha_j(D, Z) = (1-r^{(j)})I + r^{(j)} \left[b_0 Z - \frac{1}{3} (2^{2q}-1)D \right] \\ + \frac{r^{(j)}}{3} \left[(2^{2q}-1)b_0 D Z - \frac{1}{15} (2^{4q+1} - 2^{2q+3} - 2^{2q+1} + 8)D^2 \right] + O(D^2 Z + D^3). \quad (8')$$

From this expression we conclude that it is essential that at least one of the relaxation parameters equals 1.

Furthermore, by expanding $[I - b_0 Z]e^{(j-1)}$ in terms of the eigenvectors of S we see from (8) that those eigenvectors which correspond to zero eigenvalues of S will never be damped. For larger values of q , these eigenvectors are both of high and low frequency so that high degree smoothing may cause a drop in accuracy and we should not expect that the local error of the method (3) converges to the local error of the corrector. This unfavourable property of SGPC methods can be partly compensated by performing an additional iteration which has an amplification factor less than 1 for all frequencies. For instance, we may add a Jacobi iteration to the SGPC method (3) :

$$y^{(m+1)} = y^{(m)} - \frac{\omega}{1+b_0 r^{\nu} R} [y^{(m)} - b_0 r^{\nu} f(t_{n+1}, y^{(m)}) - \Sigma_n], \quad (9)$$

to obtain an SGPC method in $P(ESC)^m(EJ)E$ mode. The amplification matrix of this Jacobi iteration is given by

$$\alpha_{m+1}(Z) := I - \frac{\omega}{1+b_0 r^{\nu} R} [I - b_0 Z], \quad 0 < \omega < 2,$$

which has all its eigenvalues less than 1 in magnitude. This additional iteration does not greatly reduce the SGPC error, but it compensates the zero eigenvalues of the smoothing matrix.

We shall call $P^{(m)}(x, z)$ the *amplification polynomial* of the SGPC method since this polynomial determines the damping (or magnification) of the predictor error. Furthermore, the polynomial

$$P_m(z) := P^{(m)}(0, z) \quad (10a)$$

will be called the *generating polynomial* of the SGPC method because, given this polynomial, the amplification polynomial follows from the relation

$$P^{(m)}(x, z) = P_m \left[\frac{1}{b_0} \left[1 - \frac{2^q (1+2x)^{-1}}{2^{2q+1} x} [1 - b_0 z] \right] \right]. \quad (10b)$$

In particular, we have that the relaxation parameters $r^{(j)}$ are obtained from the zeros $z^{(j)}$ of $P_m(z)$ by means of the relation $r^{(j)} = [1 - b_0 z^{(j)}]^{-1}$.

4.2 STABILITY

As in the accuracy of SGPC methods, the amplification polynomial also plays a central role in the stability of these methods. We assume that the predictor and corrector are respectively generated by the polynomials $(\tilde{\rho}, \tilde{\sigma})$ and (ρ, σ) with $\tilde{a}_0 = a_0 = 1$. The characteristic polynomial of the PC method (3) in P(ESC)^mE mode is now given by (cf. [2])

$$C(\zeta; Z, P^{(m)}(D, Z)) = [I\rho(\zeta) - Z\sigma(\zeta)]\zeta^{\bar{k}} \quad (11a)$$

$$- [(I - b_0 Z)P^{(m)}(D, Z)][P^{(m)}(D, Z) - I]^{-1} [I\tilde{\rho}(\zeta) - Z\tilde{\sigma}(\zeta)]\zeta^k.$$

By means of the polynomial C we can define in the real (z, z^*) -plane the *stability domain*

$$\mathbb{D} := \{(z, z^*) : C(\zeta; z, z^*) \text{ has no roots outside the unit circle}\}. \quad (11b)$$

We emphasize that \mathbb{D} is completely determined by the PC pair and does not depend on $P^{(m)}(D, Z)$. In [2] plots of stability domains associated with PC pairs for first-order ODEs can be found. In the following example we give the stability domain of a family of PC pairs for second-order ODEs.

EXAMPLE 1. Consider the PC pair

$$(\tilde{\rho}(\zeta) = (\zeta - 1)^2, \tilde{\sigma}(\zeta) = 0), \quad (\rho(\zeta) = (\zeta - 1)^2, \sigma(\zeta) = \zeta + b_0(\zeta - 1)^2)$$

for integrating equations of the form $y'' = f(t, y)$. The predictor $(\tilde{\rho}, \tilde{\sigma})$ is of zero order and the corrector (ρ, σ) is of second order for all $b_0 \neq 1/12$. If $b_0 = 1/12$, then we obtain the fourth-order Numerov corrector. The characteristic polynomial $C(\zeta; z, z^*)$ is given by

$$C(\zeta; z, z^*) = \zeta^2 - \left[2 - \frac{z(z^* - 1)}{1 - b_0 z} \right] \zeta + 1.$$

From this expression it can be derived that in the left-hand part of the $(z; z^*)$ -plane the stability domain is bounded above by the line $z^* = 1$ and below by the curve $z^* = 1 - 4b_0 + 4/z$. \square

5. MODEL PROBLEM

The preceding subsections show that the amplification polynomial determines both the accuracy and stability of SGPC methods. In order to obtain criteria how we should choose the relaxation parameters in the amplification polynomial we consider a class of model problems for which the matrix Z can be presented in the form

$$Z := r^\nu \frac{\partial f(t_{n+1}, \eta)}{\partial y} = r^\nu [RD + R^* D^*]. \quad (12)$$

Here, R is the spectral radius of $\partial f/\partial y$ at (t_{n+1}, η) , R^* is a nonnegative number and D^* is a matrix with the same eigensystem as the difference matrix D and with eigenvalues in the interval $[-1, 0]$. Assuming that $R^* \ll R$ we may consider the matrix $R^* D^*$ as a *perturbation matrix*. In our earlier investigations, we always considered the case $D^* = 0$. However, in order to apply SGPC methods to strongly nonlinear problems, it is of interest to consider nonzero perturbation matrices.

Given the matrix D^* , either Z or D can be eliminated from the amplification polynomial and the stability condition (11) by means of relation (12).

5.1. THE AMPLIFICATION POLYNOMIAL

In accuracy considerations, it is convenient to express the various formulas in terms of the matrix D . From (7) it follows that the iteration error in the SGPC method (SGPC error) is given by

$$e^{(m)} = P^{(m)}(D, r^\nu [RD + R^* D^*]) e^{(0)} = \prod_{j=1}^m \alpha_j(D, r^\nu [RD + R^* D^*]) e^{(0)}. \quad (7')$$

The eigenvalues of the amplification matrix $\alpha_j(D, r^\nu [RD + R^* D^*])$ are given by

$$\alpha_j(x, r^\nu [Rx + R^* x^*]) := 1 - r^{(j)} \frac{2^q (1+2x) - 1}{2^{2q+1} x} [1 - b_0 r^\nu [Rx + R^* x^*]], \quad (8'')$$

where x and x^* run through the eigenvalues of D and D^* , respectively. It is convenient to introduce the variable

$$X = X(x, x^*) := \frac{2^q (1+2x) - 1}{2^{2q+1} x} [1 - b_0 r^\nu [Rx + R^* x^*]], \quad -1 \leq x, x^* \leq 0, \quad (13)$$

so that for the model problem (12) the amplification polynomial can be presented in the form of a polynomial $Q_m(X)$ of degree m in X :

$$P^{(m)}(x, z) = Q_m(X(x, x^*)) := \prod_{j=1}^m [1 - r^{(j)} X(x, x^*)]. \quad (14)$$

Here, we have for the moment ignored the Jacobi iteration (9). In this connection, we recall that we cannot have convergence of the SGPC error to zero because the amplification polynomial $P^{(m)}(x, z)$ equals 1 whenever $X = 0$. However, by virtue of the Jacobi iteration (9) we achieve that the corresponding eigenvector components are also damped.

The function $X(x, x^*)$ is always positive and its maximal value is assumed either at $x = 0$ or at a point $x_{\max} := (1 - \delta)[\cos(2^{-q}\pi) - 1]/2$, where δ is a small positive parameter depending on x^* . Neglecting second-order δ -terms we find

$$X(0, x^*) = 1 - b_0 r^{\nu} R^{\nu} x^*, \quad X(x_{\max}, x^*) \approx \frac{1}{4^q} [b_0 r^{\nu} R + 2 \frac{(1 - b_0 r^{\nu} R^{\nu} x^*)(1 + \delta)}{1 - \cos(2^{-q}\pi)}]. \quad (15)$$

Thus, the range of X -values is given by $[0, X_{\max}]$ with

$$X_{\max} \approx \max \left\{ 1 + b_0 r^{\nu} R^{\nu} x^*, \frac{1}{4^q} \left[b_0 r^{\nu} R + 2 \frac{(1 + b_0 r^{\nu} R^{\nu} x^*)(1 + \delta^*)}{1 - \cos(2^{-q}\pi)} \right] \right\}, \quad (16)$$

where δ^* denotes the maximum value of δ for $-1 \leq x^* \leq 0$. For small values of q , we usually have

$$X_{\max} \approx \max(1 + b_0 r^{\nu} R^{\nu} x^*, \frac{1}{4^q} b_0 r^{\nu} R), \quad (16')$$

and for larger values of q

$$X_{\max} \approx \max(1 + b_0 r^{\nu} R^{\nu} x^*, \frac{1}{4^q} b_0 r^{\nu} R + \frac{4}{\pi^2} (1 + b_0 r^{\nu} R^{\nu} x^*)(1 + \delta^*)). \quad (16'')$$

In this latter case, the first term at the right-hand side plays a crucial role. In Figure 1, the behaviour of the function $X(x, x^*)$ is plotted for $q = 4, 5$ and 6 .

Of particular interest is the location of X -values that correspond to the dominating frequencies in the predictor error. Usually, the eigenvalues of the dominating eigenvectors in $e^{(0)}$ correspond to the lowest frequencies in $e^{(0)}$, that is, to x -values close to zero. Let us consider the function $X(x, x^*)$ for small values of x (cf. (8')) :

$$X(x, x^*) \approx 1 - b_0 \tau^\nu [R x + R^* x^*] + \frac{1}{3} (4^q - 1)x.$$

From this expression and the above considerations it follows that,

if

$$b_0 \tau^\nu R > \frac{1}{3} (4^q - 1), \quad (17)$$

then, for decreasing values of x , the function $X(x, x^*)$ first increases monotonically from $X(0, x^*)$ to $X(x_{\max}, x^*)$, next it decreases from $X(x_{\max}, x^*)$ to $X = 0$, etc. If (17) is not satisfied, then $X(x, x^*)$ first decreases from $X(0, x^*)$ to $X = 0$, etc. Thus, the lowest frequencies are either directly to the left or directly to the right of $X = X(0, x^*)$ (see also Figure 1). The location of the lowest frequencies plays an important role in the construction of accurate SGPC methods.

5.2. THE REAL STABILITY BOUNDARY

Next we consider the stability of SGPC methods when applied to the model problem. Here, it is more convenient to express the stability conditions in terms of the matrix Z because the stability domain refers to Z . Since the matrices Z , D and D^* share the same eigensystem, the stability condition is given by

$$(z, z^*) \in \mathbb{D} \text{ for all pairs of eigenvalues } (z, z^*) \text{ of } \left[Z, P^{(m)} \left(\frac{Z - \tau^\nu R^* D^*}{\tau^\nu R}, Z \right) \right]. \quad (18)$$

The largest value of β such that this condition is satisfied for $0 \leq \tau^\nu R \leq \beta$ defines the *real stability boundary* of the SGPC method.

EXAMPLE 2. Consider the one-stage SGPC method generated by the PC pair given in Example 1 and set $r^{(1)} = 1$, $q \neq 0$. Then

$$P^{(1)}(z/\tau^\nu R - R^* x^*/R, z) = 1 - \frac{2^q (1 + 2(z/\tau^\nu R - R^* x^*/R)) - 1}{2^{2q+1} (z/\tau^\nu R - R^* x^*/R)} [1 - b_0 z].$$

The stability boundary is determined by the set of $\tau^2 R$ -values for which the points $(z, P^{(1)}(z/\tau^\nu R - R^* x^*/R, z))$, with $-\tau^2 R \leq z \leq 0$ and $-1 \leq x^* \leq 0$, are in the domain bounded above by the line $z^* = 1$ and bounded below by the curve $z^* = 1 - 4b_0 + 4/z$ (see Example 1). For example, it can be shown that for $R^* = 0$ the stability boundary is given by $\beta = 4^{q+1}$ for all values of b_0 . \square

If large stability boundaries are desired, then the stability domain should satisfy certain conditions. From (8) we deduce that for large negative values of z the amplification polynomial behaves as

$$P^{(m)}(z/\tau^\nu R - R^* x^*/R, z) = \prod_{j=1}^m \alpha_j(z/\tau^\nu R - R^* x^*/R, z),$$

$$\alpha_j(z/\tau^\nu R - R^* x^*/R, z) \approx 1 + r^{(j)}(1+b_0 \tau^\nu R) \frac{2^q (1+2z/\tau^\nu R) - 1}{2^{2q+1}}.$$

This implies that for large negative values of z the stability domain is required to contain the strip

$$-d_1 \leq z^* \leq 1, \quad d_1 := \prod_{j=1}^m [1 - 4^{-q} r^{(j)}(1+b_0 \tau^\nu R)]. \quad (19)$$

If the stability domain does contain such a strip, then the stability condition for SGPC methods usually reduces to prescribing an upperbound for the value of X_{\max} which leads to an explicit expression for the stability boundary. The following theorem is easily proved (cf. (16)).

THEOREM 1. Let the stability condition of the SGPC method be of the form $X_{\max} \leq c_m$, where c_m may depend on m . Then the SGPC method possesses the stability boundary

$$\beta = \frac{1}{b_0} \text{Min} \left\{ \frac{c_m - 1}{R^*/R}, c_m 4^q - 2 \frac{(1+b_0 \tau^\nu R^*)(1+\delta^*)}{1 - \cos(2^{-q}\pi)} \right\}. \quad \square \quad (20a)$$

For large values of q the stability range is approximately given by the interval $[0, \beta]$, where

$$\beta = \frac{1}{b_0} \text{Min} \left\{ \frac{c_m - 1}{R^*/R}, [c_m - \frac{4}{\pi^2} (1+b_0 \tau^\nu R^*)(1+\delta^*)] 4^q \right\}. \quad (20'a)$$

This expression shows that the stability boundary is quite substantial for small values of R^*/R and $\tau^\nu R^*$. At the same time, the stability boundary depends critically on correct estimates of R^* and δ^* unless c_m is much larger than 1. We shall call c_m the *stability constant* of the method.

In terms of the stepsize the stability condition reads

$$\tau \leq \left(\frac{\beta}{R} \right)^{1/\nu}. \quad (20b)$$

In the following subsections we discuss several choices of the amplification polynomial and the associated stability constants.

6. VARIOUS SGPC METHODS

In this section, we present various SGPC methods by considering a few natural choices of the generating polynomial $P_m(z)$. The first two subsections deal with SGPC methods in P(ESC)^mE mode, that is, without the Jacobi iteration defined by (9). In subsection 6.3, we consider the effect of the additional iteration (9).

6.1. SGPC METHODS

We first recall that conventional PC methods are obtained when we set $r^{(j)}_S = I$ in the SGPC method (3). Smoothed conventional PC methods (SCPC' methods) are obtained by choosing $r^{(j)}_S = S$, where S is defined by (6). The generating polynomial of SCPC methods is given by

$$P_m(z) = [b_0 z]^m. \quad (21)$$

According to (10) and using the variable X we can express the amplification polynomial in the form

$$P^{(m)}(x, r^\nu(Rx + R^* x^*)) = Q_m(X(x, x^*)) = [1 - X(x, x^*)]^m. \quad (22)$$

For a given problem, that is for given values of R and R^* , it is of interest to compare the behaviour of this polynomial for a small and large value of q . Choosing m fixed and the step size such that (20b) is just satisfied, i.e. $b_0 r^\nu R = b_0 \beta$, we find for large q a considerably larger maximum time step than for small values of q . However, the damping of the lower frequencies is considerably less than the damping for a small value of q . In Figure 2a and 2b, this feature is illustrated by plotting the amplification polynomial as a function of x in the interval $[-.2, 0]$. In both figures this interval corresponds to eigenvalues of the Jacobian matrix $\partial f/\partial y$ in the interval $[-.2R, 0]$. Furthermore, we see that a nonzero perturbation matrix decreases the damping power of the SCPC method.

The stability condition of SCPC methods is determined by the point where the amplification polynomial 'leaves the stability domain'. According to (19), we shall assume that the stability domain contains the strip $-D_1 \leq z^* \leq 1$, $z := r^\nu(Rx + R^* x^*) \leq 0$. For even values of m we see that $P^{(m)}(x, z)$ leaves this strip at the point where $P^{(m)}(x, z) = 1$, that is where

$X(x, x^*) = 2$. This leads us to the condition $X_{\max} \leq 2$. Similarly, for odd values of m we are led to the condition $X_{\max} \leq 1 - (-D_1)^{1/m}$. By virtue of Theorem 1 the stability boundary is given by (20a) with stability constant $c_m = 2$ for m even and $c_m = 1 - (-D_1)^{1/m}$ for m odd. Because of these relatively small c_m -values the stability boundary of SCPC methods is rather sensitive to changes in R^* and δ^* .

6.2 SSPC AND SMPG METHODS

Consider the generating polynomial

$$P_m(z) := \frac{1}{2} [d_2 - d_1 + (d_2 + d_1)T_m(w_0 + w_1(d_1, d_2)b_0 z)], \tag{23a}$$

where

$$w_1 = w_1(d_1, d_2) := T_{1/m}\left(\frac{2-d_2+d_1}{d_2+d_1}\right) - w_0, \tag{23b}$$

and where w_0 , d_1 and d_2 are free parameters (here, $T_{1/m}$ denotes the Chebyshev polynomial of fractional degree $1/m$). The amplification polynomial can be expressed in the form

$$P^{(m)}(x, \tau^\nu(Rx + R^* x^*)) = Q_m(X(x, x^*)) - \frac{1}{2} \left[d_2 - d_1 + (d_2 + d_1)T_m(w_0 + w_1(d_1, d_2)[1 - X(x, x^*)]) \right]. \tag{24}$$

This polynomial equals 1 at $X = 0$ (as it should do), it has its first zero at

$$X = \frac{T_{1/m}\left(\frac{2-d_2+d_1}{d_2+d_1}\right) - T_{1/m}\left(\frac{d_1-d_2}{d_1+d_2}\right)}{T_{1/m}\left(\frac{2-d_2+d_1}{d_2+d_1}\right) - w_0}, \tag{25}$$

and it is bounded by d_2 and $-d_1$ in the interval

$$0 \leq X \leq (1 + w_0 + w_1(d_1, d_2)) / w_1(d_1, d_2).$$

In view of our discussion of expression (8'), we choose the zero (25) at $X = 1$ so that at least one relaxation parameter equals 1. Thus,

$$w_0 = w_0(d_1, d_2) := T_{1/m}\left(\frac{d_1 - d_2}{d_1 + d_2}\right). \tag{23c}$$

One criterion for determining the remaining parameters d_1 and d_2 is the maximization of the real stability boundary. Assuming that the stability domain contains the strip $-D_1 \leq z^* \leq 1$, $z := \tau^\nu(Rx + R^* x^*) \leq 0$, we are led to

the values $d_1 = D_1$ and $d_2 = 1$ (cf. [2]). The stability boundary for these methods follows from the condition

$$X_{\max} \leq \frac{1 + w_0(D_1, 1) + w_1(D_1, 1)}{w_1(D_1, 1)},$$

that is, it is given by (20a) with stability constant

$$c_m = \frac{2}{1 - \cos\left(\frac{1}{m} \arccos \frac{D_1 - 1}{D_1 + 1}\right)} \approx \frac{4m^2}{\left[\arccos \frac{D_1 - 1}{D_1 + 1}\right]^2} \text{ as } m \rightarrow \infty. \quad (26)$$

This value is much larger than the stability constant obtained for the SCPC methods. We shall call the resulting method a *smoothed stabilized PC method* (SSPC method).

In Figure 3 the analogue of Figure 2 for SSPC methods is plotted. Figure 3a reveals that the damping of the SSPC method in the interval $[-.2, 0]$ is rather strong in spite of the fact that we did not try to minimize the magnitude of the amplification polynomial and only tried to maximize the stability boundary permitted by the condition $-D_1 \leq P^{(m)}(x, r^\nu(Rx + R^*x^*)) \leq 1$ in the region $-1 \leq x, x^* \leq 0$. However, the situation changes when the value of q increases. Then, amplification factors of magnitude 1 quickly enter the low frequency interval. This may cause a drop of accuracy. In such cases, we may try the parameter values $d_1 = d_2 = d$ where d is sufficiently small. We shall call the resulting method a *smoothed minimax PC method* (SMPC method).

In Figure 4 the analogue of the Figures 2 and 3 is plotted for SMPC methods. A comparison of the Figures 3a and 4a reveals that, in this case of second-degree generating polynomials, the damping power of the SSPC method is not much less than that of the SMPC method.

Next we turn to the stability of SMPC methods. Again assuming that the stability domain contains the strip $-D_1 \leq z^* \leq 1$, $z := r^\nu(Rx + R^*x^*) \leq 0$, the stability boundary follows from the condition

$$X_{\max} \leq \frac{1 + w_0(d, d) + w_1(d, d)}{w_1(d, d)}, \quad d \leq D_1,$$

that is, it is given by (20a) with stability constant

$$c_m = \frac{1 + T_{1/m}(\frac{1}{d})}{T_{1/m}(\frac{1}{d}) - \cos(\frac{\pi}{2m})} \approx \frac{4m^2}{[\operatorname{arccosh}(\frac{1}{d})]^2 + \frac{\pi^2}{4}} \text{ as } m \rightarrow \infty. \quad (27)$$

Although this value is smaller than the stability constant obtained for the SSPC methods, it is much larger than that of the SCPC methods.

6.3 JACOBI CORRECTION

Due to the zero eigenvalues of the smoothing matrix S , there may be eigenvector components in the predictor error that will never vanish. By adding the Jacobi iteration (9), that is, by applying the SGPC method in $P(\text{ESC})^m(\text{EJ})\text{E}$ mode, these components are to some extent removed.

In order to illustrate the effect of this iteration on the behaviour of the amplification polynomial, we have plotted in Figure 5 for the SSPC method the polynomial

$$P^{(2)}(x, r^\nu(Rx + R^* x^*)) \left[1 - \frac{\omega}{1 + b_0 r^\nu R} [1 - b_0 r^\nu (Rx + R^* x^*)] \right] \quad (28)$$

with $q = 5$, and $d_1 = 1/3$, $d_2 = 1$, $b_0 r^\nu R = 2867$ and $R^* x^*/R = 0$ for $\omega = 0$ (no Jacobi correction) and $\omega = 1$ (with Jacobi correction).

7. IMPLEMENTATION OF SSPC AND SMPG METHODS AND NUMERICAL EXPERIMENTS

7.1 IMPLEMENTATIONAL DETAILS

From an implementational point of view, it is more attractive to generate the SSPC and SMPG methods by employing the three-terms recursion satisfied by the generating polynomial, rather than to derive explicit expressions for the relaxation parameters $r^{(j)}$. It can be verified that the SSPC and SMPG methods, including the Jacobi correction iteration (9), are equivalent to the scheme :

If $m = 1$ then $y_{n+1} = y^{(0)} - SR^{(0)}$;

If $m \geq 2$ then

$$y^{(1)} = \frac{\gamma_0}{\gamma_1} F^{(0)},$$

$$\begin{aligned}
 y^{(j)} &= -\frac{\gamma_{j-2}}{\gamma_j} y^{(j-2)} + 2\frac{\gamma_{j-1}}{\gamma_j} F^{(j-1)}, \quad j = 2, \dots, m-1, \\
 y^{(m)} &= \frac{1}{2} (d_2 - d_1) y^{(0)} - \frac{1}{2} \gamma_{m-2} (d_2 + d_1) y^{(m-2)} + \gamma_{m-1} (d_2 + d_1) F^{(m-1)}, \\
 y_{n+1} &= y^{(m)} - \frac{\omega}{1 + b_0 \tau^\nu R} R^{(m)},
 \end{aligned} \tag{29}$$

where w_0 and w_1 are defined in (23c) and (23b), respectively, and where we introduced the quantities

$$\begin{aligned}
 \gamma_j &:= T_j(w_0 + w_1), \quad F^{(j)} := (w_0 + w_1) y^{(j)} - w_1 S R^{(j)}, \\
 R^{(j)} &:= y^{(j)} - b_0 \tau^\nu f(t_{n+1}, y^{(j)}) - \Sigma_n.
 \end{aligned}$$

The smoothing matrix S is defined by (6) and (4), and the method parameters d_1 and d_2 are to be chosen on the basis of the stability domain of the underlying PC pair. If the parameter ω equals zero, then no Jacobi correction is performed. If $d_2 = 1$, then all coefficients γ_j equal 1. For implementational details of the semidiscretization of the PDE at boundary points we refer to [2].

7.2. PARABOLIC PROBLEM

In all experiments, the grid used was defined by equally spaced grid points and it turned out that the time step could be chosen such that the stability condition (20) was satisfied for $R^* = \delta^* = 0$, that is, with zero perturbation matrix D^* .

In our first experiment we applied the method to the symmetric, three-point spatial discretization of the linear parabolic problem

$$u_t = u_{xx} + g(t, x), \quad 0 \leq x \leq 1, \quad 0 \leq t \leq T,$$

with source function g , initial condition and Dirichlet boundary conditions taken from the exact solution $u(t, x) = 1 + x^3 t^3$. The grid used was defined by the equally spaced grid points $x_j = j/64$ and the time step $\tau = 1/64$. In order to satisfy the stability condition we adapted the number of iterations. We tested the PC pair consisting of the linear extrapolation predictor and the second-order backward differentiation corrector. The stability domain of this PC pair requires $d_1 \leq 1/3$ and $d_2 \leq 1$. Furthermore,

the relaxation parameter ω should satisfy the condition $0 \leq \omega \leq 4/3$ in order to guarantee that the amplification polynomial stays within the stability domain.

In the tables below we present for a few values of q and ω the maximum absolute error at $t = T$ and the number N of right-hand side evaluations involved.

Table 1a. Results obtained by SSPC methods for $d_1 = 1/3$ and $d_2 = 1$ at $T = 1$.

ω	$q = 0$		$q = 1$		$q = 2$		$q = 3$		$q = 4$	
	N	error	N	error	N	error	N	error	N	error
0	882	$10^{-3.2}$	441	$10^{-3.2}$	252	$10^{-3.3}$	126	$10^{-3.3}$	63	$10^{-2.9}$
1	945	$10^{-3.3}$	504	$10^{-3.3}$	315	$10^{-3.3}$	189	$10^{-3.3}$	126	$10^{-3.3}$
4/3	945	$10^{-3.3}$	504	$10^{-3.3}$	315	$10^{-3.3}$	189	$10^{-3.3}$	126	$10^{-3.3}$

Table 1b. Results obtained by SSPC methods for $d_1 = 1/3$ and $d_2 = 1$ at $T = 10$.

ω	$q = 0$		$q = 1$		$q = 2$		$q = 3$		$q = 4$	
	N	error	N	error	N	error	N	error	N	error
0	8946	$10^{-1.9}$	4473	$10^{-2.0}$	2556	$10^{-1.7}$	1278	$10^{-1.0}$	639	$10^{-0.1}$
1	9585	$10^{-2.3}$	5112	$10^{-2.3}$	3195	$10^{-2.3}$	1917	$10^{-2.3}$	1278	$10^{-2.3}$
4/3	9585	$10^{-2.3}$	5112	$10^{-2.3}$	3195	$10^{-2.3}$	1917	$10^{-2.3}$	1278	$10^{-2.3}$

The improvement of the accuracy by performing the additional Jacobi iteration is particularly clear in the second table where the integration interval is relatively large. Instead of adding the Jacobi iteration, we can also switch to the SMPC method by decreasing the values of the parameters d_1 and d_2 . The accuracy slightly improved indeed, but this did not justify the additional work caused by a larger value of m . The accuracies obtained for $\omega \neq 0$ cannot be improved by decreasing d_1 and d_2 . This is not surprising because the error obtained by iterating the corrector to convergence is just $10^{-3.3}$ and $10^{-2.3}$ in the respective cases presented by the Tables 1a and 1b.

7.3 HYPERBOLIC PROBLEM

Again we used equally spaced grid points and the time step was chosen such that the stability condition (20) is satisfied for $R^* = \delta^* = 0$, that is, with zero perturbation matrix D^* .

We applied the method (29) with the PC pair of Example 1 to the symmetric, three-point spatial discretization of the nonlinear hyperbolic problem

$$u_{tt} = u^2 u_{xx} + g(t,x), \quad 0 \leq x \leq 1, \quad 0 \leq t \leq T,$$

with source function g and initial and Dirichlet boundary conditions taken from the exact solution $u(t,x) = e^{-t} \sin(4\pi x)$. The grid used was defined by the equally spaced grid points $x_j = j/256$, and the time step was chosen as large as allowed by the stability condition. According to Example 1 we set $d_1 = 1 - 4b_0$ and $d_2 = 1$. It turned out that performing the additional Jacobi correction iteration did not improve the accuracy, hence we set $\omega = 0$.

In the following tables, we present the maximum absolute error at the end point $T = 1$ for various values of q and b_0 , and for $m = 1$ and $m = 2$.

Table 2a. Results obtained by SSPC methods for $m = 1$ at $T = 1$.

b_0	$q = 0$		$q = 1$		$q = 2$		$q = 3$		$q = 4$		$q = 5$	
	r	error	r	error	r	error	r	error	r	error	r	error
1/12	1/256	$10^{-4.5}$	1/128	$10^{-4.8}$	1/64	$10^{-4.8}$	1/32	$10^{-4.1}$	1/16	$10^{-3.2}$	1/8	$10^{-2.8}$
1/4	1/256	$10^{-4.5}$	1/128	$10^{-4.7}$	1/64	$10^{-4.9}$	1/32	$10^{-4.1}$	1/16	$10^{-2.9}$	1/8	$10^{-2.4}$
1/2	1/256	$10^{-4.5}$	1/128	$10^{-4.7}$	1/64	$10^{-4.7}$	1/32	$10^{-4.0}$	1/16	$10^{-2.6}$	1/8	$10^{-1.5}$
1	1/256	$10^{-4.5}$	1/128	$10^{-4.5}$	1/64	$10^{-4.3}$	1/32	$10^{-3.5}$	1/16	$10^{-2.3}$	1/8	$10^{-1.0}$

Table 2b. Results by SSPC methods for $m = 2$ at $T = 1$.

b_0	$q = 0$		$q = 1$		$q = 2$		$q = 3$		$q = 4$		$q = 5$	
	r	error	r	error	r	error	r	error	r	error	r	error
1/4	1/256	$10^{-4.5}$	1/128	$10^{-4.7}$	1/64	$10^{-5.0}$	1/32	$10^{-4.8}$	1/16	$10^{-3.6}$	1/8	$10^{-2.3}$
1/2	1/150	$10^{-4.4}$	1/75	$10^{-4.4}$	1/38	$10^{-4.2}$	1/19	$10^{-3.4}$	1/10	$10^{-2.6}$	1/5	$10^{-1.3}$
1	1/138	$10^{-4.3}$	1/69	$10^{-4.2}$	1/35	$10^{-3.8}$	1/18	$10^{-2.9}$	1/9	$10^{-1.3}$	1/5	$10^{-0.1}$

In this experiment the error produced by iterating the corrector to convergence varies from $10^{-4.0}$ to $10^{-4.5}$ if the stepsize increases from 1/5 to 1/256.

REFERENCES

1. P. J. van der Houwen, C. Boon, F. W. Wubs, Analysis of smoothing matrices for the preconditioning of elliptic difference equations, *Z. Angew. Math. Mech.* 68 (1988), 3-10.
2. P. J. van der Houwen & B. P. Sommeijer, Improving the stability of predictor-corrector methods by residue smoothing, Report NM-R8707, Centre for Mathematics and Computer Science, Amsterdam (1987), submitted for publication.
3. A. Jameson, The evolution of computational methods in aerodynamics, *J. Appl. Mech.* 50 (1983), 1052-1076.
4. A. Lerat, Une class de schemas aux difference implicites pour les systemes hyperboliques de lois de conservation, *C. R. Acad. Sc. Paris, t. 288* (1979) (Serie A), 1033-1036.
5. F. G. Shuman, Numerical methods in weather prediction II. Smoothing and filtering, *Monthly Weather Review* 85 (1957), 357-371.
6. E. Turkel, Acceleration to a steady state for the Euler equations, in : *Numerical methods for the Euler equations of fluid dynamics*, SIAM Publications, Philadelphia, (1985), 218-311.
7. F. W. Wubs, Stabilization of explicit methods for hyperbolic partial differential equations, *Int. J. Numer. Methods Fluids* 6 (1986), 641-657.

P. J. van der Houwen & B. P. Sommeijer, Centre for Mathematics and Computer Science, P. O. Box 4079, 1009 AB Amsterdam, The Netherlands

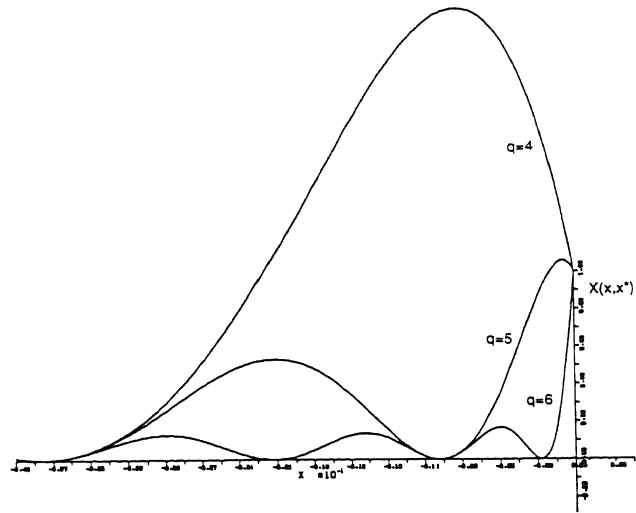


Figure 1. The function $X(x, x^*)$ for $q=4, 5, 6$ with $b_0 \tau^* R = 500$ and $R^* x^* / R = 0$.

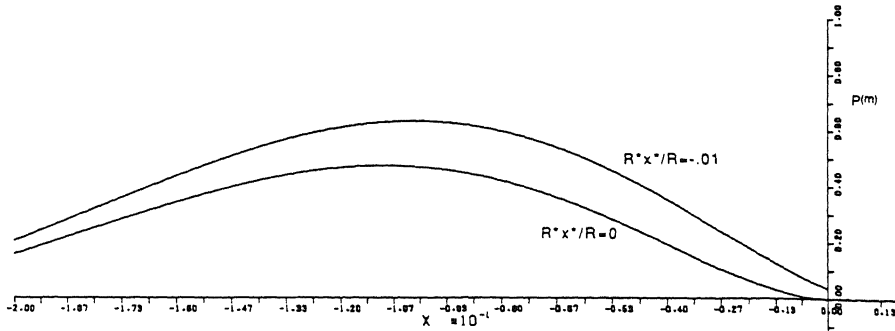


Figure 2a. The function $P^{(2)}(x, \tau^v(Rx + R^*x^*))$ for $q=2$ with $b_0\tau^v R = 19.2$, and $R^*x^*/R = 0$ and -0.01 .

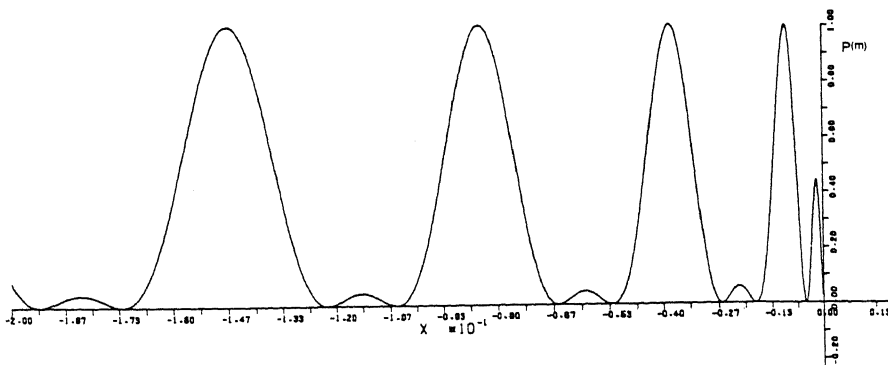


Figure 2b. The function $P^{(2)}(x, \tau^v(Rx + R^*x^*))$ for $q=5$ with $b_0\tau^v R = 1228$ and $R^*x^*/R = 0$.

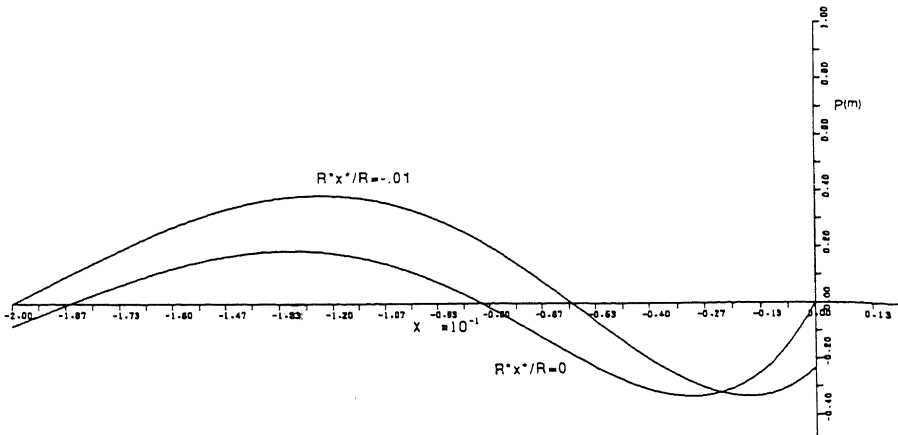


Figure 3a. The function $P^{(2)}(x, \tau^v(Rx + R^*x^*))$ for $q=2$ with $b_0\tau^vR=44.8$, and $R^*x^*/R=0$ and -0.01 .

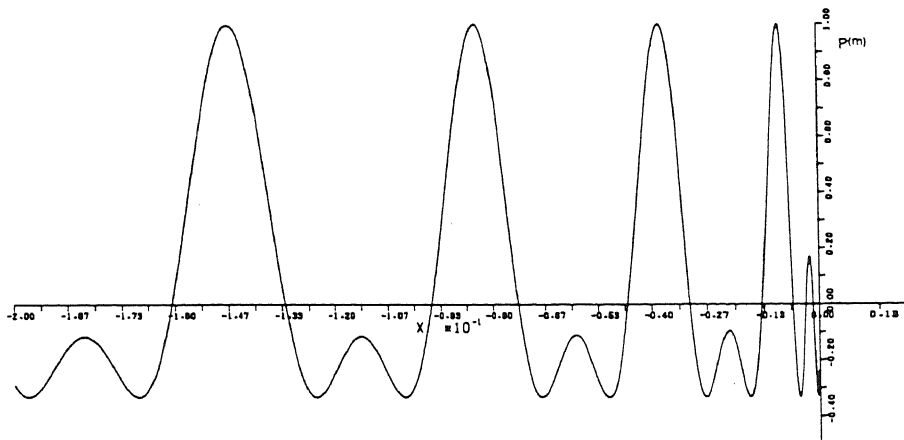


Figure 3b. The function $P^{(2)}(x, \tau^v(Rx + R^*x^*))$ for $q=5$ with $b_0\tau^vR=2867$ and $R^*x^*/R=0$.

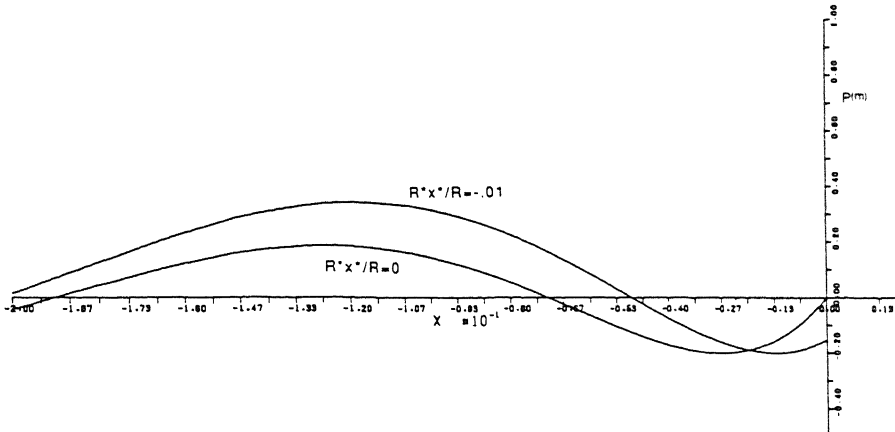


Figure 4a. The function $P^{(2)}(x, \tau^{\nu}(Rx + R^*x^*))$ for $q=2$ and $d_1=d_2=1/5$ with $b_0\tau^{\nu}R=35$ and $R^*x^*=0$ and -0.01 .

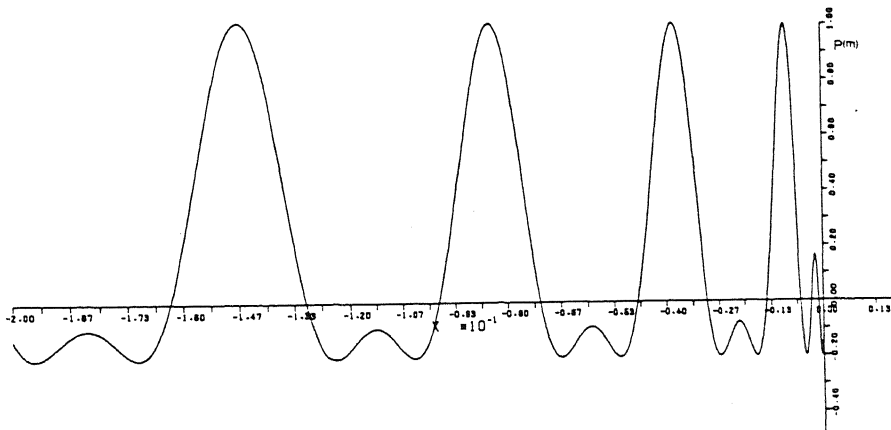


Figure 4b. The function $P^{(2)}(x, \tau^{\nu}(Rx + R^*x^*))$ for $q=5$ and $d_1=d_2=1/5$ with $b_0\tau^{\nu}R=2238$ and $R^*x^*/R=0$.

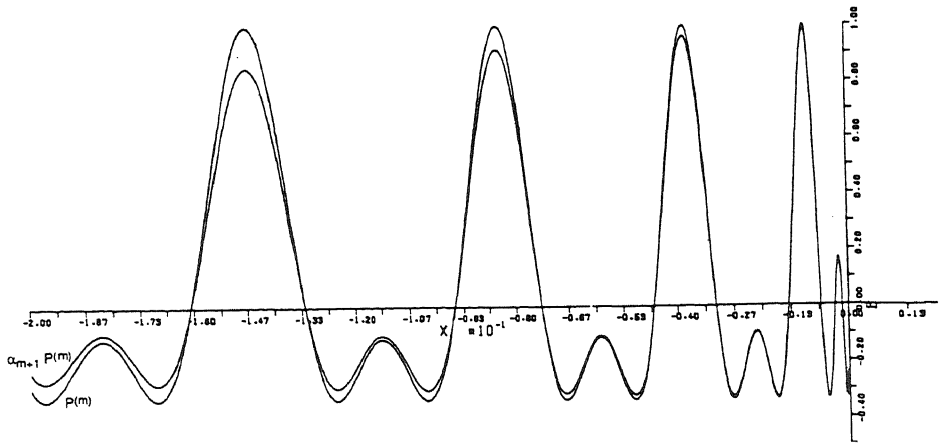


Figure 5. The function (28) for $\omega=0$ and $\omega=1$, with $q=5$, $m=2$, $d_1=1/3$, $d_2=1$, $b_0\tau^{\nu}R=2867$ and $R^*x^*/R=0$.