Sequential Regularization Methods for higher index DAEs with Constraint Singularities: I. Linear Index-2 Case

Uri Ascher*
Department of Computer Science
University of British Columbia
Vancouver, British Columbia
Canada V6T 1Z2
ascher@cs.ubc.ca

Ping Lin†
Department of Mathematics
University of British Columbia
Vancouver, British Columbia
Canada V6T 1Z2
lin@math.ubc.ca

July, 1993

Abstract

Standard stabilization techniques for higher index DAEs often involve elimination of the algebraic solution components. This may not work well if there are singularity points where the constraints Jacobian matrix becomes rank-deficient. This paper proposes instead a sequential regularization method (SRM) – a functional iteration procedure for solving problems with isolated singularities which have smooth differential solution components.

For linear index-2 DAEs we consider both initial and boundary value problems. The convergence of the SRM is described and proved in detail. Various aspects of the subsequent numerical discretization of the regularized problems are discussed as well and some numerical verifications are carried out.

*The work of this author was partially supported under NSERC Canada Grant OGP0004306.
†The work of this author was partially supported under a Killam postgraduate grant

1 Introduction

Recently more and more attention has been paid to the study of differential-algebraic equations (DAEs). Such equations arise in a variety of applications and can be difficult to solve when they have a higher index (index greater than one; cf. [9]). Higher index DAEs are ill-posed, especially when the index is greater than two [6], and a straightforward discretization generally does not work well. An alternative treatment is the use of index reduction methods, whose essence is the repeated differentiation of the constraint equations until a well-posed problem (index-1 DAEs or ODEs) is obtained. But repeated index reduction by direct differentiation leads to instability for numerical integrations (i.e. drift-off — the error in the original constraint grows). Hence, stabilized index reduction methods were proposed to overcome the difficulty. A popular stabilization technique was introduced first in the computation of constrained multibody systems by Baumgarte [7]. See [3] for corresponding improvements and additional techniques. However, for problems with constraint singularities (e.g. where some rows of the constraint Jacobian matrix become linearly dependent at some points), Bayo and Avello [8] indicate that Baumgarte's technique may not work. Another approach is the so-called regularization of DAEs where a small perturbation term (measured by a small positive parameter ϵ) is added to the original DAE (see, e.g., [11, 14, 13, 12]). The regularized problem usually is a singular perturbation problem and the DAE becomes the reduced problem of the singular perturbation problem. In the present paper, we propose a new regularization method which we call *sequential regularization method* (SRM). A motivation for this method comes from current research [8, 16] on simulation of the dynamics of constrained mechanical systems. We will show that our method works well for initial and boundary value problems with smooth solutions, with and without the presence of constraint singularities.

A singularity in the constraints (or in the algebraic solution components) of a DAE may cause various phenomena to occur, including impasse points [17] and bifurcations. In this paper, however, we assume that the solution sought is smooth in the passage through isolated singularity points, and concentrate on the linear index-2 case.

The sequential regularization method is actually a functional iteration procedure in which the difference between the exact solution of a DAE and the corresponding iterate becomes $O(\epsilon^m)$ in magnitude at the mth iteration, at least away from the starting value of the independent variable (which we shall call 'time'). Hence,

- unlike usual regularization, in the limit of the iteration we get the exact solution away from the starting time,
- the perturbation parameter ϵ does not have to be chosen very small, so the regularized problems can be less stiff and/or more stable.

As in [6], we consider a linear (or linearized) model problem

$$x^{(\nu)} = \sum_{j=1}^{\nu} A_j x^{(j-1)} + By + q, \tag{1.1a}$$

$$0 = Cx + r, \tag{1.1b}$$

where A_j, B and C are sufficiently smooth functions of $t, 0 \le t \le t_f$, $A_j(t) \in \mathbf{R}^{n_x \times n_x}, j = 1, \cdots, \nu$, $B(t) \in \mathbf{R}^{n_x \times n_y}$, $C(t) \in \mathbf{R}^{n_y \times n_x}$, $n_y \le n_x$ and CB is nonsingular (the DAE has index $\nu + 1$) except at several isolated points of t. For simplicity of exposition, let us say that there is one singularity point $t_*, 0 < t_* < t_f$. The inhomogeneities are $q(t) \in \mathbf{R}^{n_x}$ and $r(t) \in \mathbf{R}^{n_y}$. We are only interested in the case where (1.1) has a smooth solution for x. So, furthermore, we assume the following:

H1: the projector $P = B(CB)^{-1}C$ is sufficiently smooth, where we define

$$P^{(j)}(t_*) = \lim_{t \to t_*} (B(CB)^{-1}C)^{(j)}(t).$$

H2: The inhomogeneity $r(t)$ satisfies $r \in \mathcal{S}$, where

$$\mathcal{S} = \{w(t) \in \mathbf{R}^{n_y} : \text{ there exists a smooth function } z(t) \text{ s.t. } Cz = w\}.$$

We note that $H2$ is necessary for (1.1) to have a smooth solution for x and that $H1$ and $H2$ are satisfied automatically if CB is nonsingular for each t. On the other hand, neither $B(CB)^{-1}$ nor $(CB)^{-1}C$ alone are smooth near a singularity in general. Also, we note that a linearized form of the Euler-Lagrange equations which govern the motion of mechanical systems with holonomic constraints is in the form (1.1) with $\nu = 2$. Two types of singular constraints (i.e. with vanishing rows or with some rows linearly dependent at some points) mentioned in [1] both satisfy $H1$.

In this paper, we concentrate on the index-2 case, i.e. (1.1) with $\nu = 1$. For higher index cases we assume that a stable reduction to index-2 has been performed. We consider both initial and boundary value problems. In §2, we briefly discuss the conditioning of the problem with singularities. In §3, the main section of this paper, we describe the sequential regularization method and estimate its error. In §4, we consider some discretization and implementation issues for both initial and boundary value problems. Finally, several numerical examples demonstrate our theoretical results.

This paper opens up a few avenues for extension which we plan to take up with shortly. Besides the obvious desire to extend our results to nonlinear and to higher index problems, a fuller analysis complementing the results of §4 is under way and will be reported elsewhere.

2 Problem Conditioning and Baumgarte stabilization

We rewrite (1.1) with $\nu = 1$ as

$$x' = Ax + By + q, \tag{2.1a}$$
$$0 = Cx + r, \tag{2.1b}$$

and consider this DAE subject to $n_x - n_y$ boundary conditions

$$B_0 x(0) + B_1 x(t_f) = \beta\ . \tag{2.2}$$

These boundary conditions are assumed to be such that they yield a unique solution for the ODE (2.1a) on the manifold given by (2.1b). In particular, assuming for a moment that no singularities occur, if we were to replace (2.1b) by its differentiated form

$$0 = Cx' + C'x + r', \tag{2.3a}$$
$$C(0)x(0) + r(0) = 0, \tag{2.3b}$$

and use (2.3a) in (2.1a) to eliminate y and obtain n_x ODEs for x, then the boundary value problem for x with (2.2) and (2.3b) specified has a unique solution. In the initial value case $B_1 = 0$, this means that (2.2) and (2.3b) can be solved uniquely for $x(0)$. We will give a more precise assumption in Lemma 2.1 below.

Similarly to [5] and to the method of pseudo upper triangular decomposition (PUTD) described in [1] (§10.6; a difference is that we do pivoting to interchange the row with singularity of lowest order to the current row when all the other rows vanish at some singular point), there exists a smooth matrix function $R(t) \in \mathbf{R}^{(n_x - n_y)\times n_x}$, which has full row rank and satisfies

$$RB = 0, \text{ for each } t,\ 0 \le t \le t_f,$$

where R can be taken to have orthonormal rows.

As in [5, 6], define the new variable

$$u = Rx,\ 0 \le t \le t_f. \tag{2.4}$$

Then, using (2.1b), the inverse transformation is given by

$$x = Su - B(CB)^{-1}r, \tag{2.5}$$

where

$$S = (I - B(CB)^{-1}C)R^T = (I - P)R^T.$$

By the assumptions in §1, this transformation is well-defined. Differentiating (2.4) and using (2.1a) and (2.5) we obtain the essential underlying ODE (EUODE):

$$u' = (RA + R')Su - (RA + R')B(CB)^{-1}r + Rq. \tag{2.6}$$

Hence the underlying problem of (2.6) is

$$\begin{aligned} u' &= (RA + R')Su + f, && (2.7a)\\ & B_0S(0)u(0) + B_1S(t_f)u(t_f) = \beta_1. && (2.7b) \end{aligned}$$

We assume

H3: The boundary value problem (2.7) is stable, i.e. there exists a moderate-size constant K such that

$$\|u\| \le K(\|f\| + |\beta_1|),$$

where $\|u\| = max_t\{|u(t)|,\ 0 \le t \le t_f\}$.

Similarly to Theorem 2.2 of [5], we can get

Lemma 2.1 *Let the DAE (2.1) have smooth coefficients, and assume that $H1$ and $H2$ hold. If the EUODE (2.6) with the boundary conditions (2.7b) has a unique solution, then there exists a unique solution for x of problem (2.1)-(2.2) which is smooth. This implies a unique existence of a smooth By as well. Furthermore, if $H3$ holds then there is a constant K such that*

$$\|x\| \le K(\|q\| + \|B(CB)^{-1}r\| + |\beta|),$$

$$\|x'\| \le K(\|q\| + \|B(CB)^{-1}r\| + \|(B(CB)^{-1}r)'\| + |\beta|).$$

The difference between the situation here and in the nonsingular case is that here perturbation inhomogeneities r yield reasonably bounded perturbations in the solution x only if they are (in general) from the subspace *Range* (C).

From (2.1a) and (2.5), we can write

$$y = (CB)^{-1}C(x' - Ax - q), t \in [0, t_*) \cup (t_*, t_f], \tag{2.8}$$

which could be unbounded at the singular point t_* (whereas By is bounded). Note that in (2.8) C could be replaced by any appropriate matrix G with the same size as C, e.g. G can be B^T.

Remark 2.1 If B has full rank for each t, then we get

$$(CB)^{-1}C = (B^TB)^{-1}B^TP.$$

Hence, $(CB)^{-1}C$ is smooth. So, there exists a unique solution for y of problem (2.1)-(2.2) which is smooth and can be expressed as (2.8) for each t. Furthermore, using Lemma 2.1, we have in this case

$$\|y\| \le K(\|q\| + \|B(CB)^{-1}r\| + \|(B(CB)^{-1}r)'\| + |\beta|). \tag{2.9}$$

In the general case, however, we will have to consider By, rather than y alone, in the theorems of the next section. □

A Baumgarte stabilization applied to (2.1) consists in eliminating y according to (2.3),(2.8), and stabilizing. This gives the ODE

$$x' = (I - B(CB)^{-1}C)(Ax + q) - B(CB)^{-1}(C'x + r') - \epsilon^{-1}B(CB)^{-1}(Cx + r) \tag{2.10}$$

where $\epsilon > 0$ is a parameter (cf. [7, 3]). If there are no singularities then it follows from the analysis in [6] that if $H3$ holds then the boundary value problem (2.10),(2.2),(2.3b) is also stable. In other words, the "initial value stabilization" works also for the boundary value case, because the new modes introduced by replacing (2.1b) with (2.3a) are separable and decaying, in agreement with the additional *initial* conditions (2.3b).

However, in the singular case (2.10) may not work because the terms $B(CB)^{-1}C'$ and $B(CB)^{-1}r'$ are in general unbounded. Therefore, we develop in the next section an iterative method which builds up an approximation to By and x that avoids going through unbounded quantities.

3 Sequential Regularization Methods

Consider the following regularization for the DAE (2.1),

$$\epsilon x' + BE(Cx + r) = \epsilon Ax + \epsilon By + \epsilon q, \tag{3.1}$$

where $E \in \mathbf{R}^{n_y \times n_y}$ is chosen such that BEC is symmetric positive semi-definite. For example, we can choose, relying on Assumptions $H1$ and $H2$, $E = (CB)^{-1}$(hence, $BEC = P$). Using the fact that, for two matrices $M \in \mathbf{R}^{m\times n}$ and $N \in \mathbf{R}^{n\times m}$, MN and NM have the same nonzero eigenvalues, we have

$$\mathbf{Re}\lambda_i(P) = 1 > 0,\ i = 1, \cdots, n_y,\ \ \lambda_i(P) = 0,\ i = n_y + 1, \cdots, n_x. \tag{3.2}$$

Also, $E = (CB)^T$ could be a good choice in some circumstances. If CB is symmetric positive semi-definite (as in the case of mechanical systems) then it is possible to choose $E = I$, or the more typical regularization $E = (CB + \delta I)^{-1}$, with $\delta > 0$ small.

The ODE (3.1) is subject to the boundary conditions (2.2),(2.3b). This boundary value problem has a unique solution if y is given. We denote the exact solution of (2.1)-(2.2) by x_e, y_e. Obviously, if we take $y = y_e$ in (3.1) then this problem has the same solution as (2.1)-(2.2).

Because we do not know y_e in advance we propose, motivated by [8], the following iterative algorithm for solving (3.1): For $s = 1, 2, \ldots$, solve the ODE problem

$$x_s{}' = Ax_s + By_s + q \tag{3.3}$$

where

$$y_s = y_{s-1} - \frac{1}{\epsilon}E(Cx_s + r), \tag{3.4}$$

subject to the same boundary conditions (2.2) and (2.3b). Note that $y_0(t)$ is a given initial iterate and that $\epsilon > 0$ is the regularization parameter.

We call this algorithm a *sequential regularization method* (SRM). Note that $x_s(t)$ and $y_s(t)$ are defined on the entire interval $[0, t_f]$ for each s. Also, in practice we multiply (3.4) by B and keep track only of the approximations By_s to the bounded function By. If y is desired (at times other than t_*) then it can be easily retrieved from By in a post-processing step.

Remark 3.1 A penalty method for (2.1) in the initial value case (cf. [16, 12]) reads (using our notation):

$$x' = Ax + By + q, \tag{3.5a}$$

$$y = -\frac{1}{\epsilon}E(Cx + r), \tag{3.5b}$$

where $\epsilon > 0$ must be small. The regularized problem (3.5) is just like one iteration of (3.3) with $y_0(t) \equiv 0$ and $E = I$ and $B = C^T$ in [16] or $E = (CB)^{-1}$ in [12]. From (3.5b) and (2.3b), we have $y(0) = 0$. Hence, if $y_e(0) \neq 0$, the method (3.4) will not yield a good approximation for y_e near $t = 0$. In the method (3.3)-(3.4) proposed here, we can choose

$$y_0(0) = y_e(0),$$

where $y_e(0)$ can be calculated from (2.1)

$$y_e(0) = -(C(0)B(0))^{-1}((C'(0) + C(0)A(0))x_e(0) + C(0)q(0) + r'(0)).$$

□

Remark 3.2 In [16] the authors also indicated that the penalty procedure (3.5) has some drawbacks, e.g. once an error is committed in computing y, there is no compensation scheme by which the drifting of the numerical solution can be corrected. The proposed iteration (3.3),(3.4) could be such a compensation scheme since the drift of $Cx + r$ would decrease as the iteration proceeds. In [16] a so-called staggered statilization procedure is introduced for initial value problems with $B = C^T$. They first differentiate (3.5b) with $E = I$ to obtain

$$y' = -\frac{1}{\epsilon}(Cx' + C'x + r'). \tag{3.6}$$

Then, they obtain Cx' from (3.5a) and substitute it into (3.6) to yield

$$\epsilon y' + CBy = -(CA + C')x - Cq - r'. \tag{3.7}$$

The initial value for y could be the exact value. This procedure does work better for the case where CB is nonsingular if we choose ϵ to be small. But if CB is singular at t_*, or more precisely, if B is rank-deficient at t_*, this procedure may not work well since y could be unbounded as shown in (2.8). Moreover, the poor accuracy of y will also effect the accuracy of x. A similar idea appears in [12] with $E = (CB)^{-1}$, and their treatment may not work well for problems with constraint singularities either. □

Now we estimate the error of the sequential regularization method (3.3)-(3.4). We make the following assumption on the initial iterate y_0:

H4: $y_0(0) = y_e(0)$, $y_0'(0) = y_e'(0), \ldots, y_0^{(M)}(0) = y_e^{(M)}(0)$,

where M is an integer. Set $M = -1$ if $y_0(0) \neq y_e(0)$.

For initial value problems we may calculate ${y_e}^{(i)}(0)$, $i = 0, 1, \ldots$ in advance by using the ODE and its derivatives. For boundary value problem we set $M = -1$ since we don't know $y_e(0)$ beforehand.

Choosing $E = (CB)^{-1}$, we rewrite (3.4) as

$$By_s = By_{s-1} - \frac{1}{\epsilon}B(CB)^{-1}(Cx_s + r), \tag{3.8}$$

We give the following lemma before the main theorem of this section.

Lemma 3.1 *Let u, v be the solution of*

$$\begin{aligned} u' &= (RA + R')Su + S_1v + f_1, && (3.9a)\\ \delta v' + \gamma v &= \epsilon S_2 u + \epsilon S_3 v + f_2, && (3.9b)\\ B_0S(0)u(0) + B_1S(t_f)u(t_f) &= \beta - S_4v(0) - S_5v(t_f),\ v(0) = v_0, && (3.9c) \end{aligned}$$

where all coefficients are sufficiently smooth, $\delta = 1$ or $\delta = \epsilon$, γ is a positive constant and H3 holds. Then, for ϵ appropriately small or γ appropriately large, we have the following stability inequality

$$\|u\| \leq K(\|f_1\| + \|f_2\| + |\beta| + |v_0|),$$

$$\|v\| \leq K(\epsilon\|f_1\| + \|f_2\| + |\beta| + |v_0|),$$

where K is a positive constant.

Proof: Let $v = (v_1, \cdots, v_{n_y})^T$. From (3.9b), we easily have

$$|v_i| \le \frac{\epsilon}{\gamma}\|S_2\|\|u\| + \frac{\epsilon}{\gamma}\|S_3\|\|v\| + \frac{1}{\gamma}\|f_2\| + |v_0|, \; i = 1, \cdots, n_y$$

Hence, taking the maximum of the left hand side for $1 \le i \le n_y$ and choosing small ϵ or large γ appropriately such that $\epsilon\|S_3\| < \gamma$, we get

$$\|v\| \le \frac{\epsilon}{\gamma - \epsilon\|S_3\|}\|S_2\|\|u\| + \frac{\|f_2\| + \gamma|v_0|}{\gamma - \epsilon\|S_3\|}. \tag{3.10}$$

By using $H3$, from (3.9a), there exists a positive constant K_1 such that

$$\|u\| \le K_1(\|S_1\|\|v\| + \|f_1\| + |\beta| + |S_4||v(0)| + |S_5||v(t_f|)$$

$$\le K_1((\|S_1\| + |S_5|)\|v\| + \|f_1\| + |\beta| + |S_4||v_0|)$$

$$\le \frac{K_1\epsilon(\|S_1\| + |S_5|)\|S_2\|}{\gamma - \epsilon\|S_3\|}\|u\| + \frac{K_1(\|S_1\| + |S_5|)(\|f_2\| + \gamma|v_0|)}{\gamma - \epsilon\|S_3\|} + K_1(\|f_1\| + |\beta| + |S_4||v_0|).$$

Hence, by choosing smaller ϵ or larger γ such that $\frac{K_1\epsilon(\|S_1\|+|S_5|)\|S_2\|}{\gamma-\epsilon\|S_3\|} < 1$, the stability inequality for u follows. Now the stability inequality for v follows from that for u and (3.10). $\square$

We are now ready to state the main theorem of this paper.

Theorem 3.1 *Let the DAE (2.1) have sufficiently smooth coefficients, and assume that H1, H2, H3 and H4 hold. Then, for the solution of iteration (3.3),(3.8), we have the following error estimates:*

$$\begin{aligned} x_s(t) - x_e(t) &= O(\epsilon^s) + O(\epsilon^{M+2}p_s(t/\epsilon)e^{-t/\epsilon}), \\ By_s(t) - By_e(t) &= O(\epsilon^s) + O(\epsilon^{M+1}p_s(t/\epsilon)e^{-t/\epsilon}), \end{aligned}$$

for $0 \le t \le t_f$ and $s \ge 1$. Here $p_s(\tau) \equiv 0$ if $s \le M+1$; otherwise $p_s(\tau)$ is a polynomial of degree $s - M - 2$ with generic positive coefficients and $|p_s(0)| = |(By_0)^{(M+1)}(0) - (By_e)^{(M+1)}(0)|$.

Proof: Let $u_s = Rx_s$ and $w_s = Px_s$. Similarly to (2.5), we have

$$x_s = Su_s + w_s. \tag{3.11}$$

Furthermore, using (3.3) we obtain

$$\begin{aligned} u_s' &= (RA + R')Su_s + (RA + R')w_s + Rq, &\text{(3.12a)} \\ \epsilon w_s' + w_s &= \epsilon(PA + P')Su_s + \epsilon(PA + P')w_s + \epsilon By_{s-1} &\text{(3.12b)} \\ &+ \epsilon Pq - B(CB)^{-1}r, \end{aligned}$$

subject to

$$\begin{aligned} B_0 S(0) u_s(0) + B_1 S(t_f) u_s(t_f) &= \beta - B_0 w_s(0) - B_1 w_s(t_f), & \text{(3.13a)} \\ w_s(0) &= -B(0)(C(0)B(0))^{-1} r(0). & \text{(3.13b)} \end{aligned}$$

The iteration (3.8) for By becomes

$$By_s = By_{s-1} - \frac{1}{\epsilon}(w_s + B(CB)^{-1} r). \tag{3.14}$$

The proof proceeds along familiar lines of singular perturbation analysis. According to [18, 19] we can construct the asymptotic expansion of w_s and u_s sequentially for $s = 1, 2, \ldots$, where we use Lemma 3.1 to estimate the remainders. Then, using (3.14) and (3.11), we get the asymptotic expansion of By_s and x_s respectively. Note that in these expansions the first terms are exactly x_e and By_e. This process eventually yields the proof of the theorem. □

To provide a better feeling about the sequential regularization method we give in the appendix a detailed proof for the initial value case with no layers, $s \leq M + 1$. In that proof, the construction of the asymptotic expansion is directly for x and By. Moreover, the construction method we apply is somewhat different from [18, 19] and more relevant to the concept of DAEs.

Next, we consider the following sequential regularization: The same ODE (3.3) is solved successively under the same boundary conditions as before, but with the update

$$y_s = y_{s-1} - \frac{1}{\epsilon} E(\alpha_1 (Cx_s + r)' + \alpha_2 (Cx_s + r)). \tag{3.15}$$

When considering the initial value problem with $E = I$ and $B = C^T$, this corresponds to Algorithm ALF1 of [8] for constrained mechanical systems (although they do it for the corresponding index-3 case) derived by a penalty-augmented Lagrangian formulation. Bayo and Avello indicate that in multibody dynamics simulation they have used Algorithm ALF1 very successfully and that it has turned out to be very efficient and accurate. However, they have also noted that under repetitive singular conditions this algorithm may lead to unstable behaviour . For our index-2 case (2.1), it appears to be impossible to choose a matrix E such that problem (3.3),(3.15) is always stable, even if we assume $B = C^T$. A numerical example in §5 will verify such instability phenomena even for the case of one singular point. However, for the case where constraints are without singularities, (3.15) is preferable to (3.4). In particular, (3.15) yields an ODE problem for x_s which is essentially not a singular perturbation problem. Take $E = (CB)^{-1}$ as before and rewrite (3.15) as

$$By_s = By_{s-1} - \frac{1}{\epsilon} BE(\alpha_1 (Cx_s + r)' + \alpha_2 (Cx_s + r)). \tag{3.16}$$

Then we give the following error estimation for (3.3), (3.16):

Theorem 3.2 *Let the DAE (2.1) have sufficiently smooth coefficients, and assume that C has full rank and that H1,H2 and H3 hold. Then for the solution of the iteration procedure (3.3), (3.16) with $\alpha_1 \neq 0$, we have the following error estimations:*

$$x_s - x_e = O(\epsilon^s),$$

$$By_s - By_e = O(\epsilon^s)$$

for $0 \leq t \leq t_f$ and $s = 1, 2, \ldots$. Note that no boundary layer terms appear here even for $M = -1$ in H4!

Proof: Denote $u_s = Rx_s$ and $v_s = Cx_s$. Hence

$$x_s = Su_s + Fv_s, \tag{3.17}$$

where $S = (I - P)R^T$ and $F = B(CB)^{-1} = PC^T(CC^T)^{-1}$ are both sufficiently smooth. From (3.3),(3.16), we get

$$\begin{aligned} u_s' &= (RA + R')Su_s + (RA + R')Fv_s + Rq, \\ (\epsilon + \alpha_1)v_s' + \alpha_2 v_s &= \epsilon(C' + CA)Su_s + \epsilon(C' + CA)Fv_s + \epsilon CBy_{s-1} + \epsilon Cq - \alpha_1 r' - \alpha_2 r, \end{aligned}$$

with the corresponding boundary conditions, and

$$By_s = By_{s-1} - \frac{1}{\epsilon}B(CB)^{-1}(\alpha_1(v_s + r)' + \alpha_2(v_s + r)).$$

Repeating the procedure of the proof of Theorem 3.1 and using Lemma 3.1 again to estimate the remainder of the asymptotic expansion, we obtain

$$\begin{aligned} u_s - u_e &= O(\epsilon^s), \\ v_s - v_e &= O(\epsilon^s), \\ By_s - By_e &= O(\epsilon^s), \end{aligned}$$

where $u_e = Rx_e,\ v_e = Cx_e = -r$. Hence, using (3.17) and $x_e = Su_e + Fv_e$, we obtain

$$x_s - x_e = S(u_s - u_e) + F(v_s - v_e) = O(\epsilon^s).$$

□

4 Discretization and implementation issues

The SRM iteration (3.3),(3.4) (or (3.3),(3.8)) yields a sequence of ODE problems which are to be solved numerically. Multiplying (3.4) by B and inserting into (3.3), the ODE problem to be solved at the s^{th} iteration is written as the singular-singularly-perturbed problem (see [19, 15])

$$\epsilon x_s' = -BECx_s + \epsilon Ax_s + f_s \tag{4.1a}$$

$$B_0 x_s(0) + B_1 x_s(t_f) = \beta,\ C(0)x_s(0) + r(0) = 0 \tag{4.1b}$$

where $f_s(t)$ is a known inhomogeneity,

$$f_s = -BEr + \epsilon(By_{s-1} + q) \tag{4.2}$$

We consider a finite difference (or collocation) discretization of (4.1) on a mesh

$$\begin{aligned} \pi \quad : \quad & 0 = t_0 < t_1 < \ldots < t_N = t_f \\ & h_i = t_i - t_{i-1}, \qquad h = \max_{1 \le i \le N} h_i. \end{aligned}$$

and denote by x_i^s, y_i^s the corresponding approximations of $x_s(t_i)$, $y_s(t_i)$, resp. We now have essentially two small, positive parameters to choose: ϵ and h. We assume that h is chosen small enough so that the EUODE problem (2.7) may be considered as nonstiff and that the problem coefficients are sufficiently smooth. For the remaining treatment we consider initial and boundary value problems separately.

4.1 Boundary value problem (BVP)

In the BVP case the situation is the familiar one, much like the iterative solution of a nonlinear boundary value ODE using quasilinearization (see, e.g., [4]). Each of the linear boundary value ODEs (4.1) is discretized on a mesh π using, say, a symmetric finite difference scheme or some other method. If $h \ll \epsilon$ then we expect, as $h \to 0$, convergence to the solution of (4.1) and our theory then applies for the entire numerical algorithm.

For instance, suppose we use collocation at k Gaussian points per mesh interval ($k = 1$ yields the midpoint difference scheme). Denote the collocation points by

$$t_{ij} : t_{i-1} < t_{i1} < \ldots < t_{ik} < t_i$$

and require that (3.3), (3.8) hold for the collocation solution at the collocation points. With $By_{ij}^0 = By_0(t_{ij})$ and x_{ij}^s, y_{ij}^s denoting the collocation approximations of $x_s(t_{ij})$, $y_s(t_{ij})$ resp., we let

$$By_{ij}^s = By_{s-1}(t_{ij}) - \epsilon^{-1} B(t_{ij}) E(t_{ij}) (C(t_{ij}) x_{ij}^s + r(t_{ij})) \tag{4.3}$$

and obtain the collocation approximations of (4.1), (4.2). The usual theory then yields

$$x_i^s - x_s(t_i) = O(h^{2k})$$

Further observing that (3.3) is satisfied at the collocation points we obtain

$$By_{ij}^s - By_s(t_{ij}) = O(h^{k+1})$$

(see, e.g., [4]). Combining with Theorem 3.1 this yields the error estimates

$$x_i^s - x_e(t_i) = O(h^{2k} + \epsilon^s) \tag{4.4a}$$

$$By_{ij}^s - By_e(t_{ij}) = O(h^{k+1} + \epsilon^s), \ j = 1, \ldots, k \tag{4.4b}$$

away from the initial layer, i.e., for $t_i \gg \epsilon$.

A difficulty with the BVP case is that in general we do not know $y_e(0)$, hence we expect $y_0(0) \neq y_e(0)$, i.e. $M = -1$ in $H4$ and in Theorem 3.1. The two error estimates (4.4) at $t_i = O(\epsilon)$ have to be appended, therefore, by the terms $O(\epsilon e^{-t_i/\epsilon})$ and $O(e^{-t_i/\epsilon})$, respectively. This suggests that ϵ should not be taken too large: we must require

$$\epsilon \ll \min(1, t_f).$$

Once an accurate SRM solution, $\{x_i^*, B(t_i)y_i^*\}$ say, for x_e has been determined outside the initial layer, though, it may be possible to obtain an accurate solution everywhere by applying a few SRM iterations numerically solving (4.1a) (changing $-BEC$ to BEC) subject to the terminal value

$$x(t_f) = x_N^*, \tag{4.5}$$

and choosing By_0 satisfying $B(t_f)y_0(t_f) = B(t_f)y_N^*$. This procedure is feasible provided that the terminal value problem (4.1a),(4.5) is well-conditioned (which holds if the terminal value problem for the EUODE (2.7a) is well-conditioned; however, this is an additional condition to those of Lemma 2.1).

If the condition $h \ll \epsilon$ is deemed too restrictive then it is possible instead to consider taking $\epsilon \ll h_i$ for h_i away from the initial layer (i.e. for all i s.t. $t_{i-1} \gg \epsilon$). This situation has been analyzed for collocation at Gauss and Lobatto points in [2]. We recall briefly that the error estimates (4.4) are replaced by

$$\begin{aligned} x_i^s - x_e(t_i) &= O(h^{k+l} + \epsilon^s) && (4.6a) \\ By_{ij}^s - By_e(t_{ij}) &= O(h^k + \epsilon^s),\ j = 1, \ldots, k && (4.6b) \end{aligned}$$

for $t_i \gg \epsilon$, where $l = 1$ if k is odd and an additional condition on the mesh holds (see (3.13) in [2]), $l = 0$ otherwise. Note that the condition (2.25) in [2] is satisfied by the problem (4.1). Observe that the order in (4.6b) is one lower than in (4.4b), essentially because the approximation for $x(t)$ no longer has a smooth error.

4.2 Initial value problem (IVP)

For the IVP case where (4.1b) reduces to

$$x(0) = \bar{x} \text{ given} \tag{4.7}$$

we may, of course, proceed in the same way as for the BVP case. But now a few things are easier. Firstly, for this case we can calculate $By_e(0)$ and then choose By_0 to be exact at $t = 0$. In fact, as indicated earlier we can also do this for higher derivatives of By at the initial value by repeated differentiation of (2.1). Such a preparation of the initial iterate By_0 allows removing the layer error terms (or the condition $t_i \gg \epsilon$) in the error estimates above.

Secondly, one may use a more convenient difference scheme to integrate the IVP (4.1a),(4.7). If the EUODE is sufficiently nonstiff to warrant use of a nonsiff integration method then this can be an attractive possibility here. Note, though, that $-h_i/\epsilon$ must be in the absolute stability region of the method (see (3.12b)). Thus, an explicit Runge-Kutta method of order p, for instance, may necessitate (at least) p SRM iterations in order for the error in the estimates of Theorem 3.1 to be of the same order as the error in the numerical approximation. Depending on the problem one may prefer to choose ϵ smaller and apply a stiff method like BDF or collocation at Radau points (see, e.g., [10]). This again yields an advantage over the BVP case, because these methods dampen errors, unlike the symmetric difference schemes. The error estimates are the usual ones (cf. [10]). For instance, using collocation at k Radau points ($t_{ik} = t_i$; $k = 1$ yields backward Euler) we have

$$\begin{aligned} x_i^s - x_e(t_i) &= O(h^{2k-1} + \epsilon^s) && (4.8a) \\ By_{ij}^s - By_e(t_{ij}) &= O(h^k + \epsilon^s), \; j = 1, \ldots, k && (4.8b) \end{aligned}$$

for $1 \le i \le N$.

Remark 4.1 The stiffness introduced by the regularization is an IVP stiffness even in the BVP case – see the IVP for w_s in (3.12b),(3.13b). It is therefore tempting, and for many examples certainly useful, to apply a damping difference method such as Radau collocation in the above BVP context as well. However, such a method may be dangerous for a general use if the EUODE gives rise to trouble due to over-damping (see, e.g., Example 10.10 in [4]). □

The most important difference between the IVP and BVP cases is that the iterative method described here does not appear to be necessarily optimal or even natural in the IVP context, certainly not from the storage requirement point of view: Note that the entire approximation of By_{s-1} on $[0, t_f]$ needs to be stored. The situation here is similar to that encountered with other functional iteration methods like waveform methods.

However, this difficulty can be resolved by rearranging the computation, assuming that the number of the SRM iterations, m, is chosen in advance. Thus, at each time step i, $1 \le i \le N$, we calculate sequentially the quantities $x_i^1, By_i^1, x_i^2, By_i^2, \ldots, x_i^m, By_i^m$. To do this using a one-step scheme, say, we need only the corresponding quantities locally, over the mesh subinterval $[t_{i-1}, t_i)$, and By_i^0. For the latter we may use, for instance, $y_i^0 \equiv y_e(0)$, i.e. $By_i^0 = B(t_i)y_0^0$, $0 \le i \le N$. The storage requirements are now independent of N and other typical IVP techniques such as local error control may be applied as well.

5 Numerical examples

We now present a few very simple examples to demonstrate our claims in the previous sections. Throughout this section we use a constant step size h and set $t_f = 1$. To make life difficult we choose h so that there is an i such that $t_i = t_*$ (if there is a singularity). In the implementation we monitor the size of the pivot in a Gauss elimination procedure for CB and slightly perturb t_i away from t_* when needed. At a given time t, we use $'ex'$ to denote the maximum over all components of the error in x^s while $'ey'$ denotes the maximum over all components of the error in By^s. Similarly, $'drift'$ denotes the maximum residual in the algebraic equations.

We first look at a boundary value problem.

Example 1

Consider the DAE (2.1) with

$$\begin{aligned} A &= \begin{pmatrix} -1 & 1 \\ 0 & 0 \end{pmatrix}, \; B = \begin{pmatrix} 0 \\ 1-2t \end{pmatrix}, \; q = \begin{pmatrix} -\sin t \\ 0 \end{pmatrix} \\ C &= (\, 1-2t \quad 1-2t \,), \; r = -(1-2t)(e^{-t} + \sin t) \end{aligned}$$

subject to

$$x_1(1) + x_2(0) = 1/e$$

The exact solution is $x_e = (\, e^{-t} \quad \sin t \,)$, $y_e = \frac{\cos t}{1-2t}$. A singularity is located at $t = 1/2$, where y_e becomes infinite while By_e stays bounded. We start computing with the iterate $y_0(t) \equiv 0$.

In Tables 5.1 we list errors when using the midpoint scheme

$$\begin{aligned} \frac{x_i^s - x_{i-1}^s}{h_i} &= A_{i-\frac{1}{2}} x_{i-\frac{1}{2}}^s + By_{i-\frac{1}{2}}^s + q_{i-\frac{1}{2}} \\ By_{i-\frac{1}{2}}^s &= By_{i-\frac{1}{2}}^{s-1} - \epsilon^{-1} B_{i-\frac{1}{2}} E_{i-\frac{1}{2}} (C_{i-\frac{1}{2}} x_{i-\frac{1}{2}}^s + r_{i-\frac{1}{2}}) \end{aligned}$$

where $x_{i-\frac{1}{2}}^s = \frac{x_i^s + x_{i-1}^s}{2}$ (but no such relation is necessary for y^s). We apply this scheme with $h_i = h = .01$ for various values of ϵ.

Since $k = 1$, $2k = k+1 = 2$ and we expect similar $2nd$ order accuracy in ex and in ey, except for the case $\epsilon \ll h$ when the error order in By drops to 1. This is evident in the error column for $t = 1.0$. Note also the $O(\epsilon)$ improvement per SRM iteration when this term dominates the error (i.e. when $\epsilon^s \gg h^2$). Further experiments with different values of h verify the convergence orders of (4.4) and (4.6) for $k = 1$.

We note that the approximation for By at points within the initial layer is not accurate. To get a better approximation within the initial layer (i.e. near the initial point $t = 0$), we solve a terminal value problem (4.1a), (4.5), as described in §4.1.

Then we apply the SRM for the given problem with the improved values for By_0. In Table 5.2 we list the computed results after 3 iterations. They are obviously much better than the comparable ones in Table 5.1.
□

Next we consider initial value problems.

Example 2

Consider the same DAE as for Example 1 with the same exact solution but with initial values $x_1(0) = 1$, $x_2(0) = 0$ specified. From these initial conditions we can calculate $y(0) = 1$ in advance, and we choose the initial guess $y_0(t) \equiv 1$. Tables 5.3 and 5.4 display error results for $\epsilon = .1$ and $h = .001$ using the backward Euler and the forward Euler schemes, respectively. As explained in §4.2 we find all iterates at each step before proceeding to the next.

These tables show a significant improvement with each SRM iteration and no strong initial layer effect, as predicted by theory.
□

Example 3

Here we investigate the use of the modified formula (3.16) instead of (3.8) in the SRM. First, we solve the previous example numerically using (3.16). In Table 5.5 we record error values at the singularity point $t = .5$ after 3 SRM iterations, starting with $y_0(t) \equiv 1$ and using as before $\epsilon = .1$ and $h = .001$ (cf. Tables 5.3, 5.4).

From these results it is clear that the SRM with (3.16) does not work well when $\alpha_1 \neq 0$: large errors in By are obtained near the singularity and these adversely afffect the accuracy in x as well. However, the comparison changes when there is no singularity in the constraints: We now replace the constraint in Example 2 by

$$x_1 + x_2 - e^{-t} - \sin t = 0$$

leaving everything else the same (including the singularity in B). In Table 5.6 we record maximum errors in x and By over all mesh points (denote those 'exg' and 'eyg', respectively) for the starting iterates $y_0(t) \equiv 1$ and $y_0(t) \equiv 0$ (the latter does not agree with the exact $y_e(0)$).

The modified method is seen to work better for problems without singularities.
□

The above calculations all agree with our theoretical results described in Sections 3 and 4. The final example below is designed to compare our method with that of Park and Chiou [16] (i.e. (3.4) with $E = I$ for an example with $B = C^T$ and a smooth y over the entire interval).

Example 4

Consider the DAE (2.1) with

$$A = 0,\ C = (\, t - .5 \quad t^2 - .25\,)\,,\ B = C^T$$

where $q(t)$ and $r(t)$ are chosen such that the exact solution is $x_1 = x_2 = y = e^t$.

To try a wider range of values for ϵ, we use the backward Euler scheme. We set $h = .001$ and perform 4 SRM iterations with $y_0(t) \equiv 1$. This is compared to the numerical solution of (3.5a),(3.7) [16] using the same discretization. The resulting overall errors are listed in Table 5.7 and demonstrate the superiority of our approach. □

6 Appendix

To provide a better feeling about the sequential regularization method we now give a detailed proof of Theorem 3.1 for the initial value case with no layers, $s \leq M + 1$. In this proof, the construction of the asymptotic expansion is directly for x and By. Moreover, the construction method we apply is somewhat different from [18, 19] and more relevant to the concept of DAEs. For $s > M + 1$, additional initial layer expansions have to be developed. However, the construction of these layer expansions is precisely the same as in [18, 19] and so it is omitted here. In case that (2.2) are initial conditions (i.e. $B_1 = 0$) our assumptions imply that (2.2) together with (2.3b) specify $x(0)$, say

$$x(0) = \bar{x} \tag{6.1}$$

At first, consider the case $s = 1$ of (3.3),(3.8):

$$\epsilon x_1' + B(CB)^{-1}(Cx_1 + r) = \epsilon A x_1 + \epsilon B y_0 + \epsilon q,$$

with the initial conditions (6.1). This is a singular-singularly-perturbed problem (see [19, 15]). Let

$$x_1 = x_{10} + \epsilon x_{11} + \cdots + \epsilon^s x_{1s} + \cdots$$

Comparing the coefficients of like powers of ϵ, we thus have

$$\begin{aligned}
B(CB)^{-1}Cx_{10} &= -B(CB)^{-1}r && (6.2a)\\
B(CB)^{-1}Cx_{11} &= -x_{10}' + Ax_{10} + By_0 + q, && (6.2b)\\
B(CB)^{-1}Cx_{1i} &= -x_{1i-1}' + Ax_{1i-1},\ 2 \leq i \leq s + 1, && (6.2c)
\end{aligned}$$

where (6.2a) satisfies (6.1) and (6.2b) and (6.2c) satisfy homogeneous initial conditions corresponding to (6.1). Now, (6.2a) has infinitely many solutions in general. To realize the construction, we should choose x_{10} to satisfy (6.2a) and to ensure that the solution of (6.2b) exists. We choose x_{10} to be the solution x_e of problem (2.1)-(2.2), i.e.

$$\begin{aligned}
x_{10}' &= Ax_{10} + By_e + q, && (6.3a)\\
0 &= Cx_{10} + r, && (6.3b)\\
& B_0 x_{10}(0) = \beta. && (6.3c)
\end{aligned}$$

So $x_{10} = x_e$ and (6.2b) has the following form

$$B(CB)^{-1}Cx_{11} = B(y_0 - y_e). \tag{6.4}$$

Now we choose x_{11} and a corresponding y_{01} to satisfy

$$\begin{aligned} x'_{11} &= Ax_{11} + By_{01} && (6.5a)\\ Cx_{11} &= CB(y_0 - y_e), && (6.5b)\\ & B_0 x_{11}(0) = 0. && (6.5c) \end{aligned}$$

Noting that $By_e = x'_e - Ax_e - q$ is smooth, we have $CB(y_0 - y_e) \in \mathcal{S}$. Hence, using Lemma 2.1, there exists a smooth solution x_{11} of (6.5), and x_{11} satisfies (6.4). Indeed, using (6.5b) and $H4$, we have $C(0)x_{11}(0) = 0$, so $x_{11}(0) = 0$. And, from (6.5b) again,

$$(CB)^{-1}Cx_{11} = y_0 - y_e, \text{ for each } t \in [0, t_*) \cup (t_*, t_f].$$

That is,

$$B(CB)^{-1}Cx_{11} = B(y_0 - y_e), t \in [0, t_*) \cup (t_*, t_f]. \tag{6.6}$$

Taking the limit of (6.6), we thus get that x_{11} satisfies (6.4) for each $t \in [0, t_f]$.

Moreover, using $H4$, we have

$$y_{01}(0) = y'_{01}(0) = \cdots = y_{01}^{(s-1)}(0) = 0, \; s \le M + 1.$$

Also we note that By_{01} is smooth.

Generally, supposing we have got x_{1i-1}, By_{0i-1} and

$$y_{0i-1}(0) = y'_{0i-1}(0) = \cdots = y_{0i-1}^{(s-i+1)}(0) = 0$$

for $i \ge 2$, we choose x_{1i}, y_{0i} satisfying

$$x'_{1i} = Ax_{1i} + By_{0i},$$

$$Cx_{1i} = -(CB)y_{0i-1},$$

$$B_0 x_{1i}(0) = 0.$$

By the same argument as before, we obtain that x_{1i} satisfies (6.2c) for $2 \le i \le s+1$, and

$$y_{0i}(0) = y'_{0i}(0) = \cdots = y_{0i}^{(s-i)}(0) = 0, \; s \le M + 1.$$

Also, By_{0i} is smooth. Next we denote the asymptotic solution

$$\bar{x}_{1s+1} = x_{10} + \epsilon x_{11} + \cdots + \epsilon^s x_{1s} + \epsilon^{s+1} x_{1s+1}$$

and

$$z_{1s+1} = x_1 - \bar{x}_{1s+1}.$$

Then

$$\epsilon z'_{1s+1} + Pz_{1s+1} = \epsilon Az_{1s+1} + \epsilon^{s+2}(-x'_{1s+1} + Ax_{1s+1}),$$
$$z_{1s+1}(0) = 0$$

Let $u_{1s+1} = Rz_{1s+1}$ and $w_{1s+1} = Pz_{1s+1}$. Hence, we have (cf. (3.11))

$$z_{1s+1} = Su_{1s+1} + w_{1s+1}$$

and

$$u'_{1s+1} = (RA + R')Su_{1s+1} + (RA + R')w_{1s+1} + O(\epsilon^{s+1}),$$
$$\epsilon w'_{1s+1} + w_{1s+1} = \epsilon(PA + P')Su_{1s+1} + \epsilon(PA + P')w_{1s+1} + O(\epsilon^{s+2}),$$
$$u_{1s+1}(0) = 0, \ w_{1s+1}(0) = 0.$$

Using Lemma 3.1, we get $w_{1s+1} = O(\epsilon^{s+2})$ and $u_{1s+1} = O(\epsilon^{s+1})$, i.e.

$$z_{1s+1} = O(\epsilon^{s+1}).$$

Therefore,

$$x_1 = x_{10} + \epsilon x_{11} + \cdots + \epsilon^s x_{1s} + O(\epsilon^{s+1}). \tag{6.7}$$

Noting $x_{10} = x_e$, we thus obtain

$$x_1 - x_e = O(\epsilon). \tag{6.8}$$

Then, by using (3.8),(6.7),(6.2),(6.3a) and (6.5a), it follows that

$$By_1 = By_0 - \frac{1}{\epsilon}B(CB)^{-1}(Cx_1 + r)$$

$$\begin{aligned} By_1 &= By_0 - \tfrac{1}{\epsilon}(Px_{10} + B(CB)^{-1}r + \epsilon Px_{11} + \cdots + \epsilon^s Px_{1s} + O(\epsilon^{s+1})) \\ &= By_e + \epsilon By_{01} + \cdots + \epsilon^{s-1}By_{0s-1} + O(\epsilon^s) \end{aligned} \tag{6.9}$$

or

$$By_1 - By_e = O(\epsilon). \tag{6.10}$$

Now we look at the second itertation $s = 2$ of (3.3):

$$\epsilon x'_2 + B(CB)^{-1}(Cx_2 + r) = \epsilon Ax_2 + \epsilon By_1 + \epsilon q,$$

with initial conditions (6.1). Let

$$x_2 = x_{20} + \epsilon x_{21} + \epsilon^2 x_{22} + \cdots.$$

Noting that (6.9) gives us a series expansion for By_1 we obtain,

$$B(CB)^{-1}Cx_{20} = -B(CB)^{-1}r, \tag{6.11a}$$
$$B(CB)^{-1}Cx_{21} = -x'_{20} + Ax_{20} + By_e + q, \tag{6.11b}$$
$$B(CB)^{-1}Cx_{2i} = -x'_{2i-1} + Ax_{2i-1} + By_{1i-1}, \ 2 \le i \le s+1 \tag{6.11c}$$

Again, (6.11a) satisfies initial conditions (6.1) and (6.11b) and (6.11c) satisfy the corresponding homogenous ones. As the case of $s=1$, we choose $x_{20} = x_e$. We thus have

$$B(CB)^{-1}Cx_{21} = 0.$$

Then x_{21} is constructed to satisfy

$$\begin{aligned} x'_{21} &= Ax_{21} + By_{11}, && (6.12a)\\ Cx_{21} &= 0, && (6.12b)\\ & B_0x_{21}(0) = 0. && (6.12c) \end{aligned}$$

Obviously $x_{21} = 0$ since (6.12) is uniquely solvable for x_{21} by Lemma 2.1. In general, similarly to the case of $s = 1$, we choose x_{2i} satisfying

$$\begin{aligned} x'_{2i} &= Ax_{2i} + By_{1i}, && (6.13a)\\ Cx_{2i} &= -(CB)(y_{0i-1} - y_{1i-1}), && (6.13b)\\ & B_0x_{2i}(0) = 0. && (6.13c) \end{aligned}$$

for $2 \le i \le s+1$. By applying Lemma 3.1 and the same argument as in the case of $s = 1$ we get

$$x_2 = x_e + \epsilon x_{21} + \epsilon^2 x_{22} + \cdots + \epsilon^s x_{2s} + O(\epsilon^{s+1}) \qquad (6.14)$$

or

$$x_2 - x_e = O(\epsilon^2). \qquad (6.15)$$

Then, using (3.8),(6.11),(6.12a),(6.13a), (6.14) and (6.9), we conclude

$$By_2 = By_1 - \frac{1}{\epsilon}B(CB)^{-1}(Cx_2 + r) = By_e + \epsilon^2 By_{12} + \cdots + \epsilon^{s-1}By_{1s-1} + O(\epsilon^s) \quad (6.16)$$

or

$$By_2 - By_e = O(\epsilon^2) \qquad (6.17)$$

We can repeat this procedure, and, by induction, conclude the results of the theorem. □

References

[1] F. Amirouche, *Computational Methods in Multibody dynamics*, Prentice-Hall, 1992.

[2] U. Ascher, *On some difference schemes for singular singularly-perturbed boundary value problems*, Numer. Math. 46 (1985), 1-30.

[3] U. Ascher, H. Chin and S. Reich *Stabilization of DAEs and invariant manifolds*, revised manuscript, 1993.

[4] U. Ascher, R. Mattheij and R. Russell, *Numerical Solution of Boundary Value Problems for Ordinary Differential Equation*, Prentice-Hall, 1988.

[5] U. Ascher and L. Petzold, *Projected implicit Runge-Kutta methods for differential-algebraic equations*, SIAM J. Numer. Anal. 28 (1991), 1097-1120.

[6] U. Ascher and L. Petzold, *Stability of Computational Methods for Constrained Dynamics Systems*, SIAM J. Scient. Comp. 14 (1993), 95-120.

[7] J. Baumgarte, *Stabilization of constraints and integrals of motion in dynamical systems*, Comp. Math. Appl. Mech. Eng. 1 (1972), 1-16.

[8] E. Bayo and A. Avello, *Singularity-free augmented Lagrangian algorithms for constrained multibody dyneamics*, To appear in J. Nonlinear Dynamics.

[9] K. Brenan, S. Campbell and L. Petzold, *The numerical solution of higher index differential/algebraic equations by implicit Runge-Kutta methods*, SIAM J. Numer. Anal. 26 (1989).

[10] E. Hairer and G. Wanner, *Solving Ordinary Differential Equations II*, Springer-Verlag, 1991.

[11] M. Hanke, *Regularization of differential -algebraic equations revisited*, Preprint Nr. 92-19, Humboldt- Univ. Berlin, Fachbereich Mathematik, 1992.

[12] L. Kalachev and R. O'Malley, Jr., *The regularization of linear differential-algebraic equations*, Technical Report, University of Washington, 1991.

[13] L. Kalachev and R. O'Malley, Jr., *Boundary value problems for differential-algebraic equations*, manuscript, May 1993.

[14] M. Knorrenschild, *Differential/algebraic equations as stiff ordinary differential equations*, Bericht Nr. 56, Institut für Geometrie und Praktische Mathematik, RWTH Aachen, Juni 1989.

[15] R. O'Malley, Jr., *Singular Perturbation Methods for Ordinary Differential Equations*, Springer, New York, 1991.

[16] K. Park and J. Chiou, *Stabilization of computational procedures for constrained dynamical systems*, J. of Guidance 11 (1988), 365-370.

[17] P. Rabier and W. Rheinboldt, *On the computation of impasse points of quasilinear differential algebraic equations*, Technical Report ICMA-92-172, Institute for Computational Mathematics and Applications, University of Pittsburgh, 1992.

[18] A. Vasil'eva and V. Butuzov, *Asymptotic Expansion of Solutions of Singularly Perturbed Equations*, Nauka, Moscow, 1973.

[19] A. Vasil'eva and V. Butuzov, *Singularly Perturbed Equations in the Critical Case*, MRC Technical Summary Report # 2039, 1980.

ϵ	iteration	error at $\rightarrow$	t=.01	t=.1	t = .3	t=.5	t=1.0
1e-1	1	ex	.38e-1	.35e-1	.56e-1	.52e-1	.39e-1
		ey	.96	.40	.14	.34e-1	.66e-1
		drift	.87e-2	.49e-1	.51e-1	.0	.61e-1
	2	ex	.92e-2	.37e-1	.89e-2	.65e-2	.72e-2
		ey	.91	.96e-2	.14	.34e-1	.65e-2
		drift	.90e-2	.32e-1	.61e-2	.0	.72e-2
	3	ex	.94e-2	.19e-1	.12e-1	.63e-2	.15e-2
		ey	.87	.20	.30e-1	.43e-1	.85e-3
		drift	.86e-2	.16e-1	.43e-2	.0	.74e-3
5e-2	1	ex	.19e-1	.25e-1	.28e-1	.24e-1	.19e-1
		ey	.91	.15	.14	.25e-1	.32e-1
		drift	.86e-2	.34e-1	.19e-1	.0	.29e-1
	2	ex	.85e-2	.13e-1	.18e-2	.22e-2	.15e-2
		ey	.83	.15	.15e-1	.18e-2	.88e-3
		drift	.82e-2	.11e-1	.10e-3	.0	.16e-2
	3	ex	.76e-2	.80e-3	.17e-2	.23e-3	.10e-3
		ey	.75	.17	.25e-1	.16e-2	.28e-3
		drift	.74e-2	.75e-3	.43e-2	.0	.59e-4
1e-2	1	ex	.38e-2	.60e-2	.53e-2	.44e-2	.38e-2
		ey	.67	.30e-2	.40e-2	.48e-2	.62e-2
		drift	.65e-2	.80e-2	.38e-2	.0	.55e-2
	2	ex	.45e-2	.10e-3	.88e-4	.77e-4	.64e-4
		ey	.45	.32e-3	.70e-4	.44e-4	.23e-4
		drift	.44e-2	.15e-4	.14e-4	.0	.68e-4
	3	ex	.30e-2	.55e-5	.52e-5	.59e-5	.11e-4
		ey	.30	.18e-2	.26e-4	.18e-4	.67e-5
		drift	.29e-2	.17e-5	.26e-5	.0	.56e-5
1e-3	1	ex	.13e-2	.58e-3	.52e-3	.45e-3	.39e-3
		ey	.17	.47e-2	.41e-3	.49e-3	.62e-3
		drift	.17e-2	.79e-3	.38e-3	.0	.54e-3
	2	ex	.30e-3	.71e-4	.75e-5	.72e-5	.12e-4
		ey	.30e-1	.17e-1	.34e-4	.13e-4	.54e-5
		drift	.30e-3	.51e-4	.20e-5	.0	.65e-5
	3	ex	.65e-4	.15e-3	.70e-5	.70e-5	.12e-4
		ey	.70e-2	.33e-1	.12e-3	.14e-4	.56e-5
		drift	.69e-4	.11e-3	.21e-5	.0	.59e-5
1e-6	1	ex	.21e-4	.92e-5	.14e-4	.18e-4	.24e-4
		ey	.27e-2	.27e-2	.27e-2	.27e-2	.26e-2
		drift	.26e-4	.19e-5	.27e-5	.0	.18e-4
	2	ex	.19e-4	.88e-5	.14e-4	.18e-4	.24e-4
		ey	.25e-2	.25e-2	.25e-2	.25e-2	.25e-2
		drift	.18e-5	.20e-5	.28e-5	.0	.24e-4
	3	ex	.19e-4	.88e-5	.14e-4	.18e-4	.24e-4
		ey	.25e-2	.25e-2	.25e-2	.25e-2	.25e-2
		drift	.24e-4	.19e-5	.28e-5	.0	.18e-4

Table 5.1: SRM errors for Example 1 using the midpoint scheme

ϵ	error at $\rightarrow$	t=.01	t=.1	t = .3	t=.5
1e-1	ex	.62e-2	.54e-2	.39e-2	.28e-2
	ey	.48e-2	.45e-2	.44e-2	.54e-2
5e-2	ex	.76e-3	.58e-3	.32e-3	.17e-3
	ey	.22e-2	.18e-2	.10e-2	.50e-3
1e-2	ex	.57e-4	.49e-4	.35e-4	.26e-4
	ey	.10e-3	.85e-4	.52e-4	.32e-4
1e-3	ex	.49e-4	.42e-4	.31e-4	.23e-4
	ey	.56e-4	.46e-4	.30e-4	.19e-4

Table 5.2: SRM errors for Example 1 using the shooting-back technique

iteration	error at $\rightarrow$	t=.001	t=.1	t = .3	t=.5	t=1.0
1	ex	.20e-5	.72e-2	.37e-1	.63e-1	.11
	ey	.20e-2	.12	.15	.12	.59e-1
	drift	.15e-5	.60e-2	.16e-1	.76e-4	.15
2	ex	.20e-5	.51e-2	.13e-1	.10e-1	.25e-2
	ey	.20e-2	.68e-1	.45e-2	.20e-1	.80e-2
	drift	.15e-5	.42e-2	.58e-2	.14e-4	.67e-2
3	ex	.20e-5	.35e-2	.23e-2	.16e-2	.76e-3
	ey	.20e-2	.32e-1	.26e-1	.10e-1	.37e-2
	drift	.15e-5	.29e-2	.12e-2	.10e-5	.12e-2

Table 5.3: SRM errors for Example 2 using backward Euler

iteration	error at $\rightarrow$	t=.001	t=.1	t = .3	t=.5	t=1.0
1	ex	.50e-6	.71e-2	.36e-1	.63e-1	.11
	ey	.20e-2	.12	.15	.12	.60e-1
	drift	.50e-6	.60e-2	.16e-1	.76e-4	.15
2	ex	.50e-6	.51e-2	.12e-1	.10e-1	.44e-2
	ey	.20e-2	.68e-1	.41e-2	.20e-1	.70e-2
	drift	.50e-6	.42e-2	.58e-2	.14e-4	.67e-2
3	ex	.50e-6	.35e-2	.43e-2	.18e-2	.98e-3
	ey	.20e-2	.32e-1	.26e-1	.97e-2	.46e-2
	drift	.50e-6	.29e-2	.12e-2	.11e-5	.12e-2

Table 5.4: SRM errors for Example 2 using forward Euler

$(\alpha_1, \alpha_2) \rightarrow$	$(0,1)$		$(h,1)$		$(1,1)$	
method	ex	ey	ex	ey	ex	ey
backward Euler	.16e-2	.10e-1	.80e-3	.20e+1	.15	.37e+3
forward Euler	.18e-2	.96e-2	.25e-2	.18e+1	.64	.15e+4

Table 5.5: Errors near singularity using modified formula (3.15)

y_0	$(\alpha_1, \alpha_2) \rightarrow$ method	$(0,1)$ exg	$(0,1)$ eyg	$(h,1)$ exg	$(h,1)$ eyg	$(1,1)$ exg	$(1,1)$ eyg
$\equiv 1$	backward Euler	.46e-2	.44e-1	.45e-2	.43e-1	.22e-3	.28e-3
	forward Euler	.45e-2	.44e-1	.44e-2	.43e-1	.19e-3	.23e-3
$\equiv 0$	backward Euler	.22e-1	.97	.22e-1	.94	.12e-3	.75e-3
	forward Euler	.22e-1	.97	.22e-1	.94	.19e-3	.75e-3

Table 5.6: Errors for problem without singularity using modified formula (3.15)

method	$\epsilon = 10^{-1}$ exg	$\epsilon = 10^{-1}$ eyg	$\epsilon = 10^{-3}$ exg	$\epsilon = 10^{-3}$ eyg	$\epsilon = 10^{-5}$ exg	$\epsilon = 10^{-5}$ eyg
SRM (4 itns.)	.15e-2	.71e-2	.44e-4	.16e-2	.44e-4	.16e-2
Park & Chiou	.12	.45	.42e-2	.85e-1	.38e-2	.29

Table 5.7: Method comparison for Example 4