APPROXIMATING SOLUTIONS OF BOUNDARY VALUE PROBLEMS

by

Hamid Semiyari

A dissertation submitted to the faculty of
the University of North Carolina at Charlotte
in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in
Applied Mathematics

Charlotte

2015

Approved by:

_____

Dr. Douglas S. Shafer

_____

Dr. Joel D. Avrin

_____

Dr. James Sochacki

_____

Dr. Vincent Ogunro

ABSTRACT

HAMID SEMIYARI. Approximating Solutions of Boundary Value Problems.
(Under the direction of DR. DOUGLAS S. SHAFER)

We present a new algorithm for approximating solutions of two-point boundary value problems and prove theorems that give conditions under which the solution must exist and the algorithm generate approximations that converge to it. We show how to make the algorithm computationally efficient and demonstrate how the full method works both when guaranteed to do so and more broadly. We demonstrate that the method compares well against other methods commonly used in the literature. We also prove a theorem on existence of solutions of certain multi-dimensional Volterra integral equations and use it to show that the Parker-Sochacki method of introducing auxiliary variables, used to make the new algorithm computationally efficient, can be effectively applied to these Volterra integral equations in order to approximate their solutions by means of a Picard iteration scheme. Finally, we extend the existence theorem for solutions of two-point boundary value problems and prove that the new algorithm can be modified to approximate solutions in this case.

ACKNOWLEDGMENTS

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: BACKGROUND

1.1   Initial Value Problems

In the first part of this work we will be primarily interested the solution of boundary value problems associated to second order differential equations in a single dependent variable that can be placed in the form $y'' = f(t,y,y')$. It is well known that an nth order ordinary differential equation in a single dependent variable that can be solved for the nth derivative of the dependent variable can be transformed into a system of n first order ordinary differential equations. For given

$$y^{(n)} = f(t,y,y',\ldots,y^{(n-1)}), \tag{1.1}$$

where $f : \Omega \to \mathbb{R}$ for an open set $\Omega \subset \mathbb{R} \times \mathbb{R}^n$ we introduce $n-1$ additional dependent variables

$$y_1 = y', \quad y_2 = y_1', \quad \ldots, \quad y_{n-1} = y_{n-2}'$$

so that (1.1) is equivalent to

$$
\begin{aligned}
y' &= y_1 \\
y_1' &= y_2 \\
&\vdots \\
y_{n-1}' &= f(t,y,y_1,\ldots,y_{n-1})
\end{aligned}
\tag{1.2}
$$

which we write more succinctly in vector notation as

$$\mathbf{y}' = \mathbf{f}(t,\mathbf{y})$$

where

$$\mathbf{y} = \begin{pmatrix} y \\ y_1 \\ \vdots \\ y_{n-2} \\ y_{n-1} \end{pmatrix} \quad \text{and} \quad \mathbf{f}(t,\mathbf{y}) = \begin{pmatrix} y \\ y_1 \\ \vdots \\ y_{n-2} \\ f(t,y_1,\ldots,y_{n-1}) \end{pmatrix}.$$

An *initial value problem* associated to (1.2) is the system (1.2) together with the additional requirement that the solution of (1.2) satisfy the initial condition $\mathbf{y}(t_0) = \mathbf{y}_0$ for some pair $(t_0, \mathbf{y}_0) \in \Omega$. A fundamental result concerning existence and uniqueness of solutions of initial value problems is the Picard-Lindelöf Theorem. Its statement requires the following definition.

Definition 1.1.1. Let E be an open subset of $\mathbb{R}^m \times \mathbb{R}^n$. A mapping $\mathbf{f} : E \to \mathbb{R}^p : (\mathbf{y}, \mathbf{z}) \mapsto \mathbf{f}(\mathbf{y}, \mathbf{z})$ is *(uniformly) Lipschitz* on E with respect to $\mathbf{y}$ if there exists a constant L such that

$$|\mathbf{f}(\mathbf{y}, \mathbf{z}) - \mathbf{f}(\mathbf{y}', \mathbf{z})|_1 \leqslant L|\mathbf{y} - \mathbf{y}'|_2 \text{ for all } (\mathbf{y}, \mathbf{z}), (\mathbf{y}', \mathbf{z}) \in E$$

with respect to some pre-assigned norms $|\cdot|_1$ and $|\cdot|_2$. The mapping $\mathbf{f}$ is *locally Lipschitz on E with respect to* $\mathbf{y}$ if for each $(\mathbf{y}_0, \mathbf{z}_0) \in E$ there exists a neighborhood $\widetilde{E}(\mathbf{y}_0, \mathbf{z}_0)$ of $(\mathbf{y}_0, \mathbf{z}_0)$ in E on which $\mathbf{f}$ is uniformly Lipschitz with respect to $\mathbf{y}$.

Since all norms on finite dimensional vector spaces are equivalent the choice of norms in the definition is unimportant, although the constant L will depend on the specific choice of norms.

Theorem 1.1.2 (Picard-Lindelöf Theorem). *Let $I \times E$ be an open subset of $\mathbb{R} \times \mathbb{R}^n$. If $\mathbf{f} : I \times E \to \mathbb{R}^n$ is continuous and locally Lipschitz in $\mathbf{y}$ then for each pair $(t_0, \mathbf{y}_0) \in I \times E$ there exists a solution to the initial value problem*

$$\mathbf{y}' = \mathbf{f}(t, \mathbf{y}), \quad \mathbf{y}(t_0) = \mathbf{y}_0 \tag{1.3}$$

on an open interval $J \subset I$ about t_0, and any two such solutions agree on their common domain about t_0.

There are two popular methods for proving the theorem, both based on the fact that the initial

value problem (1.3) is equivalent to the integral equation

$$\mathbf{y}(t) = \mathbf{y}_0 + \int_{t_0}^{t} \mathbf{f}(s, \mathbf{y}(s)) \, ds \tag{1.4}$$

(see Lemma 1.3.1 for the precise statement). In the Picard iteration approach to the proof of the theorem, a sequence of mappings is defined by $\mathbf{y}_0(t) \equiv \mathbf{y}_0$ and

$$\mathbf{y}_k(t) = \mathbf{y}_0 + \int_{t_0}^{t} \mathbf{f}(s, \mathbf{y}_{k-1}(s)) \, ds \quad \text{for} \quad k \geqslant 1$$

and is shown to converge to a solution of (1.4). In the contraction mapping approach to the proof, the right hand side of (1.4) is used to define a mapping on a suitable function space, the mapping is shown to be a contraction, and thus the mapping has a unique fixed point, which is of course a solution of (1.4). Both of these ideas will figure into the proof of the main theorem in Chapter 2.

The algorithm that will be developed in Chapter 2 for the solution of certain boundary value problems is based on a kind of Picard iteration as just described. It is well known that in many instances Picard iteration performs poorly in actual practice, even for relatively simple differential equations. A great improvement can sometimes be had by exploiting ideas originated by G. Parker and J. Sochacki [?], which we will now describe. We begin with the following example, which illustrates the ideas involved.

Example 1.1.3. Consider the initial value problem

$$y' = \sin y, \quad y(0) = \pi/2. \tag{1.5}$$

The function on the right hand side of this autonomous differential equation is globally Lipschitz on $\mathbb{R}$ with Lipschitz constant $L = 1$, hence by the Picard-Lindelöf Theorem there exists a unique solution to the initial value problem defined on some interval J about $t = 0$. To approximate it by means of Picard iteration, from the integral equation that is equivalent to (1.5), namely,

$$y(t) = \frac{\pi}{2} + \int_{0}^{t} \sin y(s) \, ds$$

we define the Picard iterates

$$y_0(t) \equiv \tfrac{\pi}{2}, \quad y_{k+1}(t) = \tfrac{\pi}{2} + \int_0^t \sin y_k(s)\, ds \quad \text{for } k \geqslant 0.$$

The first few iterates are

$$y_0(t) = \tfrac{\pi}{2}$$
$$y_1(t) = \tfrac{\pi}{2} + t$$
$$y_2(t) = \tfrac{\pi}{2} - \cos(\tfrac{\pi}{2} + t)$$
$$y_3(t) = \tfrac{\pi}{2} + \int_0^t \cos(\sin s)\, ds,$$

the last of which cannot be expressed in closed form. However, we can avoid this impasse by introducing additional dependent variables originally so as to eliminate the transcendental function of the dependent variable that appears on the right hand side of the differential equation in (1.5). Specifically, we define variables u and v by $u = \sin y$ and $v = \cos y$ so that $y' = u$, $u' = (\cos y)y' = uv$, and $v' = (-\sin y)y' = -u^2$, hence the original one-dimensional problem is embedded as the first component in the three-dimensional problem

$$
\begin{aligned}
y' &= u & y(0) &= \tfrac{\pi}{2} \\
u' &= uv & u(0) &= 1 \\
v' &= -u^2 & v(0) &= 0.
\end{aligned}
\tag{1.6}
$$

Indeed, let the unique solution to (1.5) be $y = \sigma(t)$ on some interval J about 0 and let the unique solution to (1.6) be $(y, u, v) = (\rho(t), \mu(t), \nu(t))$ on some interval K about 0. By construction $(y, u, v) = (\sigma(t), \sin\sigma(t), \cos\sigma(t))$ solves (1.6) on J, hence we conclude that on $J \cap K$ the function $y = \sigma(t)$, which in the general case we cannot find explicitly, is equal to the function $y = \rho(t)$, which we can approximate on any finite interval about 0 to any required accuracy. For although the dimension has increased, now the right hand sides of the differential equations are all polynomial functions so

quadratures can be done easily. In particular, the Picard iterates are now

$$\mathbf{y}_0(t) = \begin{pmatrix} \frac{\pi}{2} \\ 1 \\ 0 \end{pmatrix} \qquad \mathbf{y}_{k+1}(t) = \mathbf{y}_0 + \int_0^t \mathbf{y}_k(s)\,ds = \begin{pmatrix} \frac{\pi}{2} \\ 1 \\ 0 \end{pmatrix} + \int_0^t \begin{pmatrix} u_k(s) \\ u_k(s)v_k(s) \\ -u_k^2(s) \end{pmatrix} ds.$$

The first component of the first few iterates are

$$y_0(t) = \frac{\pi}{2}$$

$$y_1(t) = \frac{\pi}{2} + t$$

$$y_2(t) = \frac{\pi}{2} + t$$

$$y_3(t) = \frac{\pi}{2} + t - \frac{1}{6}t^3$$

$$y_4(t) = \frac{\pi}{2} + t - \frac{1}{6}t^3 + \frac{1}{24}t^5$$

$$y_5(t) = \frac{\pi}{2} + t - \frac{1}{6}t^3 + \frac{1}{24}t^5$$

$$y_6(t) = \frac{\pi}{2} + t - \frac{1}{6}t^3 + \frac{1}{24}t^5 - \frac{61}{5040}t^7$$

The exact solution to the problem (1.5) is $y = \sigma(t) = 2\arctan(e^t)$, whose Maclaurin series is

$$\sigma(t) = \frac{\pi}{2} + t - \frac{1}{6}t^3 + \frac{1}{24}t^5 - \frac{61}{5040}t^7 + O(t^9).$$

It is apparent that the iterative process is in fact generating the Maclaurin series of the first component of the solution of problem (1.6) (which is shown in [?] to hold in general; see Theorem 1.1.4 below).

In practice we are not always able to obtain a closed form expression for the limit of the iterates $y_n(t)$, but must settle for an approximation y_n of the actual solution $y(t)$. However, obtaining an approximate solution with a sufficiently large n is practical since the algorithm is computationally efficient.

In general the Parker-Sochacki method is an approach to obtaining approximate solutions of systems of ordinary differential equations. As the example illustrates, the idea is to introduce variables that are equal to various non-polynomial functions that appear in the system of differential

equations, as well as variables that are equal to their derivatives, in such a way that those functions are expressible as solutions of a system of differential equations with polynomial right hand sides. Moreover, the initial conditions in the original problem force the initial values of the new variables. Thus we obtain a polynomial initial value problem and, by an argument based on uniqueness of solutions of initial value problems just like that given in Example 1.1.3, the solution to the original initial value problem is one component. The process guarantees that the iterative integral in Picard iteration applied to the new, larger system is easy to compute, since the integrand is always a polynomial function. The method produces Maclaurin series solutions to systems of differential equations, with the coefficients in either algebraic or numerical form. The following theorem is stated and proved in [?].

Theorem 1.1.4. *Let $\mathcal{F} = (f_1, \cdots, f_n) : \mathbb{R}^n \to \mathbb{R}^n$ be a polynomial and $\mathbf{y} = (y_1, \cdots, y_n) : \mathbb{R} \to \mathbb{R}^n$. Consider initial value problem*

$$y_j' = f_k(\mathbf{y}), \qquad y_j(0) = \alpha_j, \quad j = 1, \cdots, n$$

and the corresponding Picard iterates

$$P_{j,1}(t) = \alpha_j, \quad j = 1, \cdots, n$$
$$P_{j,k+1}(t) = \alpha_j + \int_0^t f_j(P_k(s)) \, ds, \quad k = 1, 2, \cdots, \quad j = 1, \cdots, n.$$

Then $P_{j,k+1}$ is the kth Maclaurin Polynomial for y_j plus a polynomial all of whose terms have degree greater than k. (Here $P_k(s) = (P_{1,k}(s), \cdots, P_{n,k}(s))$.)

In [?] the authors address the issue as to which systems of ordinary differential equations can be handled by this method. The procedure for defining the new variables is neither algorithmic nor unique. However, with sufficient ingenuity it has been successfully applied in every case for which the original differential equation is analytic. The following example further illustrates the method, but in a much more complicated situation.

Example 1.1.5. Consider the initial value problem

$$y'' = e^{\cos y} \sin t \left(\sqrt{y'}\right)^3, \qquad y(0) = \alpha, \quad y'(0) = \gamma > 0, \tag{1.7}$$

where the condition on γ insures that there will be a unique solution on an interval about $t = 0$. As usual we form the equivalent system of first order ordinary differential equations by introducing a new dependent variable $x = y'$, obtaining

$$y' = x$$
$$x' = e^{\cos y} \sin t \left(\sqrt{y'} \right)^3 . \tag{1.8}$$

For the term $e^{\cos y}$ in the x' equation, if we introduce $u = e^{\cos y}$ then $u' = e^{\cos y}(-\sin y)y'$ and $(\sin y)' = (\cos y)y'$, which prompts us to introduce $v = \sin y$ and $w = \cos y$, so that system (1.8) is now

$$y' = x, \quad x' = u \sin t\, x^{\frac{3}{2}}, \quad u' = -uvx, \quad v' = wx, \quad w' = -vx.$$

If $r = \sin t$, $r' = \cos t$ so we introduce $s = \cos t$ to obtain

$$y' = x, \quad x' = urx^{\frac{3}{2}}, \quad u' = -uvx, \quad v' = wx, \quad w' = -vx, \quad r' = s, \quad s' = r.$$

Finally, to deal with the term $(\sqrt{x})^3$ we introduce $\mu = \sqrt{x}$. Then $\mu' = \frac{1}{2}x^{-\frac{1}{2}}x'$ so we introduce $\rho = 1/\sqrt{x} = \mu^{-1}$, for which $\rho' = -\mu^{-2}\mu' = -\rho^2 \frac{1}{2}\rho ur\mu^3$ and ultimately obtain the system

$$
\begin{aligned}
y' &= x & y(0) &= \alpha \\
x' &= ur\mu^3 & x(0) &= \gamma \\
u' &= -uvux & u(0) &= e^{\cos \alpha} \\
v' &= wx & v(0) &= \sin \alpha \\
w' &= -vx & w(0) &= \cos \alpha \\
r' &= s & r(0) &= 0 \\
s' &= -r & s(0) &= 1 \\
\mu' &= \tfrac{1}{2}\rho ur\mu^3 & \mu(0) &= \sqrt{\gamma} \\
\rho' &= -\tfrac{u}{r}\rho^3\mu^3 & \rho(0) &= 1/\sqrt{\gamma}
\end{aligned}
\tag{1.9}
$$

which is well defined since $\gamma > 0$. We may now apply Picard iteration to (1.9) without danger of

encountering impossible quadratures, obtaining a longer and longer initial segment of the Maclaurin series of the first component $y(t)$ of the solution, which is the solution of (1.7).

1.2 Boundary Value Problems

A *two-point boundary value problem* associated to (1.2) on $\Omega = I \times E$ is the system (1.2) together with the additional requirement that the solution of (1.2) exist on an interval $(a,b) \subset I$ and satisfy a boundary condition $g(\mathbf{y}(a), \mathbf{y}(b)) = 0$ for some mapping g from E into $\mathbb{R}^n$. Our chief concern will be with problems of the form

$$y'' = f(t, y, y'), \qquad y(a) = \alpha, \quad y(b) = \beta. \tag{1.10}$$

In contrast with initial value problems of this form, regardless of the regularity of f the problem (1.10) can have either no solutions, infinitely many solutions, or a unique solution. Consider, for example, a boundary value problem of the form

$$y'' = ay' + by, \qquad y(0) = \alpha, \quad y(h) = \beta. \tag{1.11}$$

Since all solutions of the differential equation in (1.11) are known explicitly, it is not difficult to verify that this boundary value problem has a unique solution if and only if either (i) $a^2 + 4b \geqslant 0$ or (ii) $a^2 + 4b < 0$ and $h\sqrt{-a^2 - 4b} \neq 2k\pi$ for any $k \in \mathbb{Z}$. In the remaining cases it has no solution if (among other possibilities) exactly one of α and β is zero and has infinitely many solutions if (among other possibilities) $\alpha = \beta = 0$.

A relatively general theorem that guarantees that problem (1.10) will possess a unique solution is the following. It is a special case of Theorem 1.2.2 of [?], where a proof is given.

Theorem 1.2.1. *Suppose the function $f(t, u, v) = f(t, \mathbf{u})$ in problem (1.10) is defined on the set $R = \{(t, \mathbf{u}) : a \leqslant t \leqslant b\} \subset \mathbb{R} \times \mathbb{R}^2$ and satisfies*

a. f is Lipschitz in $\mathbf{u} = (u, v)$,

b. f is continuously differentiable, and

c. throughout R the first partial derivatives of f with respect u and v satisfy, for some constant M,

$$\frac{\partial f}{\partial u} > 0 \qquad and \qquad \left| \frac{\partial f}{\partial v} \right| \leqslant M.$$

Then the boundary problem (1.10) *has a unique solution.*

When we specialize to *linear* second order two-point boundary value problems we have the following well-known result (see, for example, [**?**]).

Theorem 1.2.2. *The second-order linear boundary value problem*

$$y'' = p(t)y' + q(t)y + r(t), \qquad y(a) = \alpha, \quad y(b) = \beta \qquad (1.12)$$

has a unique solution provided

a. $p(t)$, $q(t)$, *and* $r(t)$ *are continuous on* $[a,b]$ *and*

b. $q(t) > 0$ *on* $[a,b]$.

The unique solution can be decomposed as

$$y(t) = y_1(t) + \left(\frac{\beta - y_1(b)}{y_2(b)} \right) y_2(t) \qquad (1.13)$$

where $y_1(t)$ *is the unique solution of the initial value problem*

$$y'' = p(t)y' + q(t)y + r(t), \qquad y(a) = \alpha, \quad y'(a) = 0 \qquad (1.14)$$

and $y_2(t)$ *is the unique solution of the initial value problem*

$$y'' = p(t)y' + q(t)y, \qquad y(a) = 0, \quad y'(a) = 1. \qquad (1.15)$$

Proof. The existence and uniqueness of the solution of (1.12) is an immediate consequence of Theorem 1.2.1.

As to the decomposition, as a preliminary let $y_0(t)$ denote the unique solution of (1.12) and, as indicated in the statement of the theorem, let $y_1(t)$ and $y_2)$ denote the unique solutions of the intial value problems (1.14) and (1.15), respectively, which exist by the Picard-Lindelöf Theorem 1.1.2. Let $\widehat{y}_2(t)$ denote *any* solution (including $y_2(t)$) of

$$y'' = p(t)y' + q(t)y, \qquad y(a) = 0,$$

that is, of (1.15) but with the condition on the derivative at a omitted. Then obviously the function $y_3(t) \stackrel{\text{def}}{=} y_0(t) + \widehat{y}_2(t)$ solves the differential equation in (1.12) and satisfies $y_3(a) = \alpha + 0 = \alpha$. If additionally $\widehat{y}_2(b) = 0$ then $y_3(b) = \beta + 0 = \beta$ so that $y_3(t)$ is then a solution of (1.12) as well as $y_0(t)$, hence by uniqueness of solutions $\widehat{y}_2(t) \equiv 0$. Thus $y_2(b)$ cannot be zero, since $y_2(t)$ is certainly not identically zero, so the function given by (1.13) is well-defined. It obviously solves (1.12).

1.3 Equivalences, Norms, and Contraction Mappings

In this section we collect for reference some known results that will be needed later. We begin with two equivalences between initial value problems and integral equations.

Lemma 1.3.1. *Suppose I is an open interval in $\mathbb{R}$, $t_0 \in I$, E is an open set in $\mathbb{R}^n$, and $\mathbf{f} : I \times E \to \mathbb{R}^n$ is a continuous mapping.*

a. If J is an open subset of I that contains t_0 and $\eta : J \to E$ is continuous and satisfies

$$\eta(t) = \mathbf{y}_0 + \int_{t_0}^{t} \mathbf{f}(s, \eta(s)) \, ds \tag{1.16}$$

then η is differentiable and solves the initial value problem

$$\mathbf{y}' = \mathbf{f}(t, \mathbf{y}), \quad \mathbf{y}(t_0) = \mathbf{y}_0. \tag{1.17}$$

b. Conversely, if J is as in part (a) and $\eta : J \to E$ is differentiable and solves (1.17) then the integral in (1.16) exists and η satisfies (1.16).

The second equivalence is also known, but is less familiar, so we will include a sketch of the proof.

Lemma 1.3.2. *Suppose I is an open interval in $\mathbb{R}$, $t_0 \in I$, E is an open set in $\mathbb{R}^2$, and $f : I \times E \to \mathbb{R}$ is a continuous function.*

a. If h is a positive real number and η is a twice continuously differentiable function on an open neighborhood of $[t_0, t_0 + h] \subset I$ that solves the second order initial value problem

$$y'' = f(t, y, y'), \qquad y(t_0) = \alpha, \quad y'(t_0) = \gamma \tag{1.18}$$

then η solves the integral equation

$$y(t) = \alpha + \gamma(t - t_0) + \int_{t_0}^{t} (t - s) f(s, y(s), y'(s)) \, ds. \tag{1.19}$$

b. *Conversely, if η is a continuously differentiable function on an open neighborhood of $[t_0, t_0 + h]$ that solves (1.19) then η is twice continuously differentiable on an open neighborhood of $[t_0, t_0 + h]$ and solves (1.18).*

Proof. For part (a), suppose η is as hypothesized and $s \in [t_0, t_0 + h]$. Integrating (1.18) from t_0 to s yields

$$\eta'(s) - \eta'(t_0) = \int_{t_0}^{s} f(u, \eta(u), \eta'(u)) \, du$$

which when integrated from t_0 to t yields

$$\eta(t) - \eta(t_0) - \eta'(t_0)(t - t_0) = \int_{t_0}^{t} \left[\int_{t_0}^{s} f(u, \eta(u), \eta'(u)) \, du \right] ds.$$

Reversing the order of integration of the iterated integrals on the right hand side gives

$$\eta(t) - \eta(t_0) - \eta'(t_0)(t - t_0) = \int_{t_0}^{t} \left[\int_{u}^{t} f(u, \eta(u), \eta'(u)) \, ds \right] du$$

from which we obtain

$$\eta(t) = \eta(t_0) + \eta'(t_0)(t - t_0) + \int_{t_0}^{t} (t - u) f(u, \eta(u), \eta'(u)) \, du.$$

Conversely, if $h > 0$ and η is a continuously differentiable function on an open neighborhood of $[t_0, t_0 + h]$ that solves (1.19) then differentiating (1.19) gives

$$\eta'(t) = \gamma + (t - t) f(t, \eta(t), \eta'(t)) \cdot 1 + (t - t_0) f(t_0, \eta(t_0), \eta'(t_0)) \cdot 0$$
$$+ \int_{t_0}^{t} \tfrac{\partial}{\partial t} [(t - s) f(s, \eta(s), \eta'(s))] \, ds \tag{1.20}$$
$$= \gamma + \int_{t_0}^{t} f(s, \eta(s), \eta'(s)) \, ds.$$

Since f, η, and η′ are continuous the right hand side is differentiable, we may differentiate again to obtain $\eta''(t) = f(t, \eta(t), \eta'(t))$, and by (1.19) and (1.20) η also satisfies the initial conditions. $\square$

Two norms that we will use are the following.

Definition 1.3.3.

a. The *sum norm* on $\mathbb{R}^2$ is the norm defined by

$$\left|\begin{pmatrix} u \\ v \end{pmatrix}\right|_{\text{sum}} = |u| + |v|.$$

b. The *supremum norm* (or just *sup norm*) on the set $\mathcal{C}$ of bounded continuous functions from a specified subset A (usually an interval) in $\mathbb{R}$ into $\mathbb{R}^2$, with respect to the sum norm on $\mathbb{R}^2$, is the norm defined by

$$|\eta|_{\text{sup}} = \sup\{|\eta(t)|_{\text{sum}} : t \in A\}.$$

We will also need the concept of a contraction and the Contraction Mapping Theorem.

Definition 1.3.4. Let X be a vector space with norm $||\cdot||$. A *contraction mapping* on X is a mapping $\mathcal{T} : X \to X$ for which there exists a constant c with $0 \leqslant c < 1$ such that for all x and y in X

$$||\mathcal{T}(x) - \mathcal{T}(y)|| \leqslant c||x - y||. \tag{1.21}$$

Theorem 1.3.5 (Contraction Mapping Theorem)**.** *If $\mathcal{T}$ is a contraction mapping on a commmplete normed vector space X then there exists a unique fixed point of $\mathcal{T}$ in X, that is, a unique element x_0 of X such that $\mathcal{T}(x_0) = x_0$. Moreover for any $x \in X$ the sequence $\mathcal{T}^n(x) = (\mathcal{T} \circ \cdots \circ \mathcal{T})(x)$ (n-fold composition) converges to x_0.*

CHAPTER 2: APPROXIMATING SOLUTIONS OF BOUNDARY VALUE PROBLEMS

In this chapter we present a new algorithm for approximating solutions of two-point boundary value problems and prove a theorem that gives conditions under which it is guaranteed to succeed. We show how to make the algorithm computationally efficient and demonstrate how the full method works both when guaranteed to do so and more broadly. In the first section the original idea and its application are presented. It is illustrated with a number of examples in the second section. In the third section we show how to modify the same basic idea and procedure in order to apply it to problems with boundary conditions of mixed type. In the fourth section a different approach to problems treated in the third section is given which can perform better in practice.

2.1 An Efficient Algorithm for Approximating Solutions of Two-Point Boundary Value Problems

Consider a two-point boundary value problem of the form

$$y'' = f(t,y,y'), \qquad y(a) = \alpha, \quad y(b) = \beta. \tag{2.1}$$

If f is locally Lipschitz in the last two variables then by the Picard-Lindelöf Theorem, Theorem 1.1.2 (applied to (2.4) below), for any $\gamma \in \mathbb{R}$ the initial value problem

$$y'' = f(t,y,y'), \qquad y(a) = \alpha, \quad y'(a) = \gamma \tag{2.2}$$

will have a unique solution on some interval about $t = a$. Introducing the variable $u = y'$ we obtain the first order system that is equivalent to (2.2),

$$y' = u$$
$$u' = f(t,y,u)$$
$$y(a) = \alpha$$
$$u(a) = \gamma$$

$$\tag{2.3}$$

or more succinctly, writing $\mathbf{y} = (y, u) \in \mathbb{R}^2$, $\mathbf{y}_0 = (\alpha, \gamma)$, and $\mathbf{f} : \mathbb{R} \times \mathbb{R}^2 \to \mathbb{R}^2 : (t, \mathbf{y}) \mapsto (u, f(t, \mathbf{y}))$,

$$\mathbf{y}' = \mathbf{f}(t, \mathbf{y})$$
$$\mathbf{y}(a) = \mathbf{y}_0,$$

(2.4)

which by Lemma 1.3.1 is equivalent to

$$\mathbf{y}(t) = \mathbf{y}(a) + \int_a^t \mathbf{f}(s, \mathbf{y}(s)) \, ds.$$

(2.5)

The Picard iterates based on (2.5) are known to converge to the unique solution of (2.2).

The boundary value problem (2.1) will have a solution if and only if there exists $\gamma \in \mathbb{R}$ such that (i) the maximal interval of existence of the unique solution of (2.2) contains the interval $[a, b]$, and (ii) the unique solution $y(t)$ of (2.2) satisfies $y(b) = \beta$. But we can identify such a γ by applying Lemma 1.3.2 to (2.2) to obtain the equivalent integral equation

$$y(t) = \alpha + \gamma(t - a) + \int_a^t (t - s) f(s, y(s), y'(s)) \, ds,$$

(2.6)

which we then solve, when evaluated at $t = b$, for γ:

$$\gamma = \frac{1}{b-a} \left(\beta - \alpha - \int_a^b (b - s) f(s, y(s), y'(s)) \, ds \right)$$

(2.7)

The key idea in the new method is that in the Picard iteration scheme based on (2.5) we use (2.7) to also iteratively obtain successive approximations to the value of γ in (2.2), if it exists, using some initial choice $\gamma^{[0]}$ of γ, say $\gamma^{[0]} = (\beta - \alpha)/(b - a)$, the average slope of the solution to (2.1) on the interval $[a, b]$. Thus the iterates are

$$y^{[0]}(t) \equiv \alpha$$
$$u^{[0]}(t) \equiv \frac{\beta - \alpha}{b - a}$$
$$\gamma^{[0]} = \frac{\beta - \alpha}{b - a}$$

(2.8a)

and

$$y^{[k+1]}(t) = \alpha + \int_a^t u^{[k]}(s)\,ds$$

$$u^{[k+1]}(t) = \gamma^{[k]} + \int_a^t f(s,y^{[k]}(s),u^{[k]}(s))\,ds \tag{2.8b}$$

$$\gamma^{[k+1]} = \frac{1}{b-a}\left(\beta - \alpha - \int_a^b (b-s)f(s,y^{[k]}(s),u^{[k]}(s))\,ds\right).$$

This gives the following algorithm for approximating solutions of (2.1), in which we have changed the update of γ in (2.8) to incorporate it into the update of u. (See Section 2.3 for a similar application of these ideas to problems with boundary conditions of mixed type.)

Algorithm 2.1.1. To approximate the solution of the boundary value problem

$$y'' = f(t,y,y'), \qquad y(a) = \alpha, \quad y(b) = \beta \tag{2.9}$$

iteratively compute the sequence of functions on $[a,b]$

$$y^{[0]}(t) \equiv \alpha$$
$$u^{[0]}(t) \equiv \frac{\beta - \alpha}{b-a} \tag{2.10a}$$

and

$$y^{[k+1]}(t) = \alpha + \int_0^t u^{[k]}(s)\,ds$$

$$u^{[k+1](t)} = \frac{1}{b-a}\left(\beta - \alpha - \int_0^h (h-s)f(y^{[k]}(s),u^{[k]}(s))\,ds\right) + \int_0^t f(y^{[k]}(s),u^{[k]}(s))\,ds. \tag{2.10b}$$

We will now state and prove a theorem that gives conditions guaranteeing that the problem (2.9) has a unique solution, then prove that the iterates in Algorithm 2.1.1 converge to it. We will need the following technical lemma.

Lemma 2.1.2. *Let $E \subset \mathbb{R} \times \mathbb{R}^2$ be open and let $f : E \to \mathbb{R} : (t,y,u) \mapsto f(t,y,u)$ be Lipschitz in $\mathbf{y} = (y,u)$ on E with Lipschitz constant L with respect to absolute value on $\mathbb{R}$ and the sum norm on $\mathbb{R}^2$. Then*

$$\mathbf{F} : E \to \mathbb{R}^2 : (t,y,u) \mapsto (u, f(t,y,u))$$

is Lipschitz in $\mathbf{y}$ with Lipschitz constant $1 + L$ with respect to the sum norm on $\mathbb{R}^2$.

Proof. For $(t, \mathbf{y}_1)$ and $(t, \mathbf{y}_2)$ in E,

$$
\begin{aligned}
|\mathbf{F}(t,\mathbf{y}_1) - \mathbf{F}(t,\mathbf{y}_2)|_{\text{sum}} &= \left| \begin{pmatrix} u_1 \\ f(t,y_1,u_1) \end{pmatrix} - \begin{pmatrix} u_2 \\ f(t,y_2,u_2) \end{pmatrix} \right|_{\text{sum}} \\
&= |u_1 - u_2| + |f(t,y_1,u_1) - f(t,y_2,u_2)| \\
&\leqslant |u_1 - u_2| + |y_1 - y_2| + L|\mathbf{y}_1 - \mathbf{y}_2|_{\text{sum}} \\
&= |\mathbf{y}_1 - \mathbf{y}_2|_{\text{sum}} + L|\mathbf{y}_1 - \mathbf{y}_2|_{\text{sum}} \\
&= (1+L)|\mathbf{y}_1 - \mathbf{y}_2|_{\text{sum}}. \quad \square
\end{aligned}
$$

Theorem 2.1.3. *Let $f : [a,b] \times \mathbb{R}^2 \to \mathbb{R} : (t,y,u) \mapsto f(t,y,u)$ be Lipschitz in $\mathbf{y} = (y,u)$ with Lipschitz constant L with respect to absolute value on $\mathbb{R}$ and the sum norm on $\mathbb{R}^2$. If $0 < b - a < (1 + \frac{3}{2}L)^{-1}$ then for any $\alpha, \beta \in \mathbb{R}$ the boundary value problem*

$$
y'' = f(t,y,y'), \qquad y(a) = \alpha, \quad y(b) = \beta \tag{2.11}
$$

has a unique solution.

Proof. By Lemma 1.3.1 a twice continuously differentiable function η from a neighborhood of $[a,b]$ into $\mathbb{R}$ solves the ordinary differential equation in (2.11) if and only the mapping $(y(t),u(t)) = (\eta(t), \eta'(t))$ from that neighborhood into $\mathbb{R}^2$ solves the integral equation (1.16),

$$
\begin{pmatrix} y(t) \\ u(t) \end{pmatrix} = \begin{pmatrix} y(0) \\ u(0) \end{pmatrix} + \int_a^t \begin{pmatrix} u(s) \\ f(s,y(s),u(s)) \end{pmatrix} ds.
$$

The discussion leading up to (2.7) shows that η meets the boundary conditions in (2.11) if and only if $\eta(a) = \alpha$ and $\eta'(a) = \gamma$ where γ is given by (2.7), now with $(y(t),u(t)) = (\eta(t),\eta'(t))$. In short the boundary value problem (2.11) is equivalent to the integral equation

$$
\begin{pmatrix} y(t) \\ u(t) \end{pmatrix} = \begin{pmatrix} \alpha \\ \frac{1}{b-a}\left[\beta - \alpha - \int_a^b (b-s)f(s,y(s),u(s))\,ds \right] \end{pmatrix} + \int_a^t \begin{pmatrix} u(s) \\ f(s,y(s),u(s)) \end{pmatrix} ds. \tag{2.12}
$$

If $\mathbf{y}(t) = (y(t),u(t))$ is a bounded continuous mapping from a neighborhood U of $[a,b]$ in $\mathbb{R}$ into

$\mathbb{R}^2$ then the right hand side of (2.12) is well defined and defines a bounded continuous mapping from U into $\mathbb{R}^2$. Thus letting C denote the set of bounded continuous mappings from a fixed open neighborhood U of $[a,b]$ into $\mathbb{R}^2$, a twice continuously differentiable function η on U into $\mathbb{R}$ solves the boundary value problem (2.11) if and only if $\boldsymbol{\eta} \overset{\text{def}}{=} (\eta, \eta')$ is a fixed point of the operator $\mathscr{T} : C \to C$ defined by

$$\mathscr{T}\begin{pmatrix} y \\ u \end{pmatrix}(t) = \begin{pmatrix} \alpha \\ \frac{1}{b-a}\left[\beta - \alpha - \int_a^b (b-s)f(s,y(s),u(s))\,ds\right] \end{pmatrix} + \int_a^t \begin{pmatrix} u(s) \\ f(s,y(s),u(s)) \end{pmatrix} ds,$$

which we abbreviate to

$$\mathscr{T}(\mathbf{y})(t) = \begin{pmatrix} \alpha \\ \frac{1}{b-a}\left[\beta - \alpha - \int_a^b (b-s)f(s,\mathbf{y}(s))\,ds\right] \end{pmatrix} + \int_a^t \mathbf{F}(s,\mathbf{y}(s))\,ds \qquad (2.13)$$

by defining $\mathbf{F} : [a,b] \times \mathbb{R}^2 \to \mathbb{R}$ by $\mathbf{F}(t,y,u) = (u, f(t,y,u))$.

The vector space C equipped with the supremum norm is well known to be complete. Thus by the Contraction Mapping Theorem, Theorem 1.3.5, the theorem will be proved if we can show that $\mathscr{T}$ is a contraction on C. To this end, let $\boldsymbol{\eta}$ and $\boldsymbol{\mu}$ be elements of C. Let $\varepsilon = \max\{t - b : t \in U\}$. Then for any $t \in U$

$$|(\mathscr{T}\boldsymbol{\eta})(t) - (\mathscr{T}\boldsymbol{\mu})(t)|_{\text{sum}}$$

$$\leqslant \left| \begin{pmatrix} \alpha \\ \frac{1}{b-a}\left[\beta - \alpha - \int_a^b (b-s)f(s,\boldsymbol{\eta}(s))ds\right] \end{pmatrix} - \begin{pmatrix} \alpha \\ \frac{1}{b-a}\left[\beta - \alpha - \int_a^b (b-s)f(s,\boldsymbol{\mu}(s))ds\right] \end{pmatrix} \right|_{\text{sum}}$$

$$+ \left| \int_a^t \mathbf{F}(s,\boldsymbol{\eta}(s)) - \mathbf{F}(s,\boldsymbol{\mu}(s))\,ds \right|_{\text{sum}}$$

$$\leqslant \frac{1}{b-a}\int_a^b (b-s)|f(s,\boldsymbol{\eta}(s)) - f(s,\boldsymbol{\mu}(s))|\,ds + \int_a^t |\mathbf{F}(s,\boldsymbol{\eta}(s)) - \mathbf{F}(s,\boldsymbol{\mu}(s))|_{\text{sum}}\,ds$$

$$\overset{(1)}{\leqslant} \frac{1}{b-a}\int_a^b (b-s)L|\boldsymbol{\eta}(s) - \boldsymbol{\mu}(s)|_{\text{sum}}\,ds + \int_a^t (1+L)|\boldsymbol{\eta}(s) - \boldsymbol{\mu}(s)|_{\text{sum}}\,ds$$

$$\leqslant \frac{1}{b-a}\int_a^b (b-s)L||\boldsymbol{\eta} - \boldsymbol{\mu}||_{\text{sup}}\,ds + \int_a^t (1+L)||\boldsymbol{\eta} - \boldsymbol{\mu}||_{\text{sup}}\,ds$$

$$\leqslant \left(\frac{1}{b-a}\int_a^b (b-s)L\,ds + (1+L)((b-a)+\varepsilon) \right) ||\boldsymbol{\eta} - \boldsymbol{\mu}||_{\text{sup}}$$

$$= [\tfrac{1}{2}(b-a)L + (1+L)((b-a)+\varepsilon)]||\boldsymbol{\eta} - \boldsymbol{\mu}||_{\text{sup}}$$

where for inequality (1) Lemma (2.1.2) was applied in the second summand.

Thus $||\mathscr{T}\boldsymbol{\eta} - \mathscr{T}\boldsymbol{\mu}||_{\sup} \leqslant (1 + \frac{3}{2}L)((b-a) + \varepsilon))||\boldsymbol{\eta} - \boldsymbol{\mu}||_{\sup}$ and $\mathscr{T}$ is a contraction provided $(1 + \frac{3}{2}L)((b-a) + \varepsilon) < 1$, equivalently, provided $(1 + \frac{3}{2}L)(b-a) < 1 - (1 + \frac{3}{2}L)\varepsilon$. But U can be chosen arbitrarily, hence $(1 + \frac{3}{2}L)\varepsilon$ can be made arbitrarily small, giving the sufficient condition of the theorem. $\quad\square$

The second statement in the Contraction Mapping Theorem (Theorem 1.3.5) and the fact that repeated composition of the mapping $\mathscr{T}$ in the proof of the theorem generates Picard iterates guarantees that the iterates defined by (2.10) will converge to the unique solution of (2.11) that the theorem guarantees to exist. Thus we have the following result.

Theorem 2.1.4. *Let $f : [a,b] \times \mathbb{R}^2 \to \mathbb{R} : (t,y,u) \mapsto f(t,y,u)$ be Lipschitz in $\mathbf{y} = (y,u)$ with Lipschitz constant L with respect to absolute value on $\mathbb{R}$ and the sum norm on $\mathbb{R}^2$. If $0 < b - a < (1 + \frac{3}{2}L)^{-1}$ then for any α, $\beta \in \mathbb{R}$ the iterates generated by Algorithm 2.1.1 converge to the unique solution of the boundary value problem*

$$y'' = f(t,y,y'), \qquad y(a) = \alpha, \quad y(b) = \beta \tag{2.14}$$

guaranteed by Theorem 2.1.3 to exist.

Remark 2.1.5. The examples in the following section will show that the use of Algorithm 2.1.1 is by no means restricted to problems for which the hypotheses of Theorem 2.1.4 are satisfied. It will in fact give satisfactory results for many problems that do not satisfy those hypotheses.

We note that because the convergence is in the supremum norm at any step in the algorithm the approximation is a function that is uniformly very close to the exact solution on the entire interval of interest. We also remark than when Algorithm 2.1.1 is applied to a problem whose solution is known, in order to study its performance, it can be convenient to keep track of the updated values of the approximations $\gamma^{[k]}$ of γ, which means using the iteration (2.8) in place of the iteration (2.10) stated in the algorithm.

When the right hand side of the equation (2.14) is a polynomial function then the integrations that are involved in implementing Algorithm 2.1.1 can always be done, and done efficiently. But when the right hand side of equation (2.14) is not a polynomial then in a manner analogous to what

happened with equation (1.5) the Picard iterates can lead to impossible integrations. In this situation we use the auxiliary variable method of Parker and Sochacki in conjunction with Algorithm 2.1.1 to obtain a computationally efficient method for approximating the solution. Here are the details.

Supposing that the right hand side in (2.14) is not a polynomial in y and y', in the usual way introduce the new dependent variable $u = y'$ to obtain the equivalent system

$$
\begin{aligned}
y' &= u \\
u' &= f(t, y, u).
\end{aligned}
\tag{2.15}
$$

find functions $h_1(t, y, u), \ldots, h_r(t, y, u)$ and polynomials $P_0, P_1, \ldots, P_r$ in $r + 3$ indeterminates such that

(i) $f(t, y, u) = P_0(t, y, u, h_1(t, y, u), \ldots, h_r(t, y, u))$ and

(ii) if $(y(t), u(t))$ solves (2.15) then $v_j(t) = h_j(t, y(t), u(t))$ solves $v'_j = P_j(t, y, u, v_1, \ldots, v_r)$, $v_j(t_0) = h_j(t_0, y_0, u_0)$, $1 \leqslant j \leqslant r$.

$$
\mathbf{y}' = \mathbf{f}(t, \mathbf{y}),
\tag{2.16}
$$

where

$$
\mathbf{y} = \begin{pmatrix} y \\ u \\ v_1 \\ \vdots \\ v_r \end{pmatrix}
\quad \text{and} \quad
\mathbf{f}(t, \mathbf{y}) = \begin{pmatrix} u \\ P_0(t, y, u, v_1, \ldots, v_r) \\ P_1(t, y, u, v_1, \ldots, v_r) \\ \vdots \\ P_r(t, y, u, v_1, \ldots, v_r) \end{pmatrix},
$$

If γ is the value at $t = a$ of the solution of the boundary value problem (2.14) then we will still have expression (2.7) for γ but with $f(t, y, y')$ replaced by $P_0(t, y, u, v_1, \ldots, v_r)$. Assuming the value of $(v_j)_0 \stackrel{def}{=} h_1(a, \alpha, \frac{\beta - \alpha}{b - a})$ is forced by all j by the original initial conditions $y(a) = \alpha$ and $y'(a) = \gamma$, we have the usual initial value problem, the first component $y(t)$ of whose solution solves the original boundary value problem (2.14).

We can proceed as before with picard iterates for (2.16), updating the estimate for γ at each step. For although the dimension has increased, now the right hand sides of the differential equations are all polynomial functions so quadratures can be done easily. Specifically, the initialization is

$$y^{[0]}(t) \equiv \alpha$$

$$u^{[0]} \equiv \frac{\beta - \alpha}{b - a}$$

$$v_1(t) \equiv (v_1)_0$$

$$\vdots \qquad \vdots$$

$$v_r(t) \equiv (v_r)_0$$

$$\gamma^{[0]} \equiv \frac{\beta - \alpha}{b - a}$$

and the recursion is

$$
\begin{pmatrix} y^{[k+1]}(t) \\ u^{[k+1]}(t) \\ v_1^{[k+1]}(t) \\ \vdots \\ v_r^{[k+1]}(t) \end{pmatrix} = \begin{pmatrix} \alpha \\ \gamma^{[k]} \\ (v_1)_0 \\ \vdots \\ (v_r)_0 \end{pmatrix} + \int_a^t \begin{pmatrix} u^{[k]}(s) \\ P_0(s, y^{[k]}(s), u^{[k]}(s), v_1^{[k]}(s), \dots, v_r^{[k]}(s))) \\ P_1(s, y^{[k]}(s), u^{[k]}(s), v_1^{[k]}(s), \dots, v_r^{[k]}(s)) \\ \vdots \\ P_r(s, y^{[k]}(s), u^{[k]}(s), v_1^{[k]}(s), \dots, v_r^{[k]}(s)) \end{pmatrix} ds
$$

$$\gamma_{k+1} = \frac{1}{b-a} \left[\beta - \alpha - \int_a^b (b-s) P_0(s, y^{[k]}(s), u^{[k]}(s), v_1^{[k]}(s), \dots, v_r^{[k]}(s)) \, ds \right].$$

(2.17)

As before we can change the update of the estimate of γ to incorporate it into the update of $u^{[k]}(t)$.

2.2   Examples

In this section we implement the method to solve some linear and nonlinear problem and compare the result with the exact solution. We also show how the method behaves for problems with no solutions or with infinity many solutions.

Example 2.2.1. The second order linear two-point boundary value problem

$$y'' = -y, \qquad y(0) = 1, \quad y(\tfrac{\pi}{4}) = \sqrt{2}$$

has the unique solution

$$y(t) = \cos t + \sin t = 1 + t - \frac{1}{2!}t^2 - \frac{1}{3!}t^3 + \frac{1}{4!}t^4 + \frac{1}{5!}t^5 + O(t^6).$$

Since $f(t,y,y') = -y$ has Lipschitz constant 1, Theorem 2.1.4 guarantees existence of a unique solution on intervals of length up to $\frac{2}{5}$, but we will see that the algorithm works well for the much longer interval in this example. Using the recursion formulas (2.8) in Algorithm 2.1.1 instead of (2.10) so that we have a record of the ending value of γ, the iterates are

$$y^{[k+1]} = 1 + \int_0^t u^{[k]} ds$$

$$u^{[k+1]} = \gamma[k] - \int_0^t y^{[k]} ds$$

$$\gamma^{[k+1]} = \frac{4}{\pi}\left(\sqrt{2} - 1 + \int_0^{\frac{\pi}{4}} \left(\frac{\pi}{4} - s\right)y^{[k+1]} ds\right).$$

Maple code for $n = 5$ iterations is

```
restart;
n := 5;
a := 0;
b := Pi/4;
alpha := 1;
beta  := sqrt(2);
h := b - a;
y[0] := alpha;
gamma[0] := (beta-alpha)/h;
u[0] := gamma[0];

for k from 0 to n-1 do
y[k+1] := alpha+int(u[k], t = a .. t);
u[k+1] := gamma[k]-int(y[k], t = a .. t);
gamma[k+1] := (beta-alpha+int((b-t)*y[k+1],t=a..b))/h;
end:

Exact:=cos(t)+sin(t);
Error := abs(evalf(y[n]-Exact));
plot(Error, t = 0 .. h);
N:=degree(y[n+1],t);
```

The exact solution $y(t)$, the approximations $y^{[k]}(t)$, and the approximations $\gamma^{[k]}$ to the exact

value $\gamma = 1$, with coefficients rounded to five decimal places, are

$\gamma^{[0]} = 0.92009,\ \gamma^{[1]} = 0.97431,\ \gamma^{[2]} = 0.99450,\ \gamma^{[3]} = 0.99840,\ \gamma^{[4]} = 0.99965,\ \gamma^{[5]} = 0.99990,$

which are tending monotonically to the exact value $\gamma = 1$ and

$$y^{[0]} = 1$$

$$y^{[1]} = 1 + 0.52739t$$

$$y^{[2]} = 1 + 0.92009t - 0.50000t^2$$

$$y^{[3]} = 1 + 0.97431t - 0.50000t^2 - 0.08790t^3$$

$$y^{[4]} = 1 + 0.99450t - 0.50000t^2 - 0.15335t^3 + 0.04167t^4$$

$$y^{[5]} = 1 + 0.99840t - 0.50000t^2 - 0.16239t^3 + 0.04167t^4 + 0.00439t^5$$

$$y = 1 + 1.00000t - 0.50000t^2 - 0.16667t^3 + 0.04167t^4 + 0.00833t^5$$

A plot of the error in the final approximation $y^{[5]}(t)$ is shown in Figure 2.1. The qualitative features of the plot are as expected, since we are forcing agreement of $y^{[k]}(0)$ with α, we expect the Maclaurin polynomial agreement to fall off as t moves away from $t = 0$, but then the approximation improves approaching $t = b$ because agreement of $y^{[k]}(b)$ with β is forced to be good by the increasingly accurate approximation of γ by $\gamma^{[k]}$.

The maximum error of the fifth approximation is $||y - y^{[5]}||_{\sup} \approx 0.00037$. When the number of iterations is increased from five to eight the error plot looks qualitatively the same but the maximum error drops dramatically to about 5.6×10^{-5}.

Example 2.2.2. This example illustrates the effect on the rate of convergence by examining a boundary value problem with the same differential equation and the same solution as in Example 2.2.1 but on an interval twice as long:

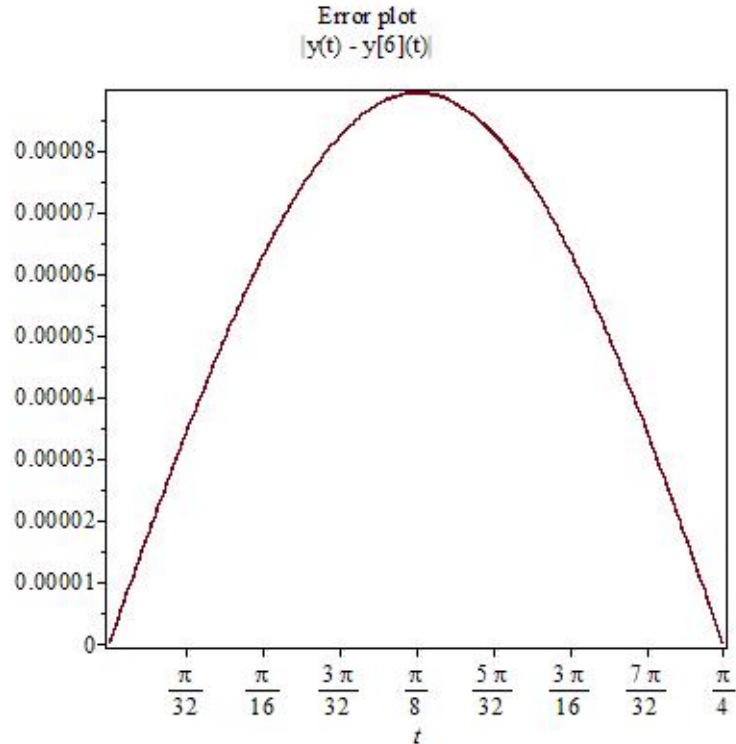$$y'' = -y, \qquad y(0) = 1, \quad y(\pi/2) = 1.$$

Figure 2.1: Error plot $|y_{\text{exact}}(t) - y^{[5]}(t)|$ for Example 2.2.1.

After five iterations we obtain, rounding coefficients to five decimal places, $\gamma = 0.98674$ and

$$y^{[5]} = 1 + 0.94689t - 0.50000t^2 - 0.13090t^3 + 0.04167t^4 - 0.00000t^5.$$

The error plot looks qualitatively the same as that in Figure 2.1 but the maximum error is now $||y - y^{[5]}||_{\text{sup}} \approx 0.023$. When the number of iterations is increased from five to eight the maximum error drops to about 0.0023.

Example 2.2.3. In this example we keep the same ordinary differential equation but choose boundary conditions for which there is no solution:

$$y'' = -y, \qquad y(0) = 1, \quad y(\pi) = 1.$$

As the number n of iterations is increased the last function computed, $y^{[n]}(t)$, tends to blow up away from the endpoints $a = 0$ and $b = \pi$ in the sense that $||y^{[n]}||_{\text{sup}}$ becomes increasingly large. For example, with $n = 5$, $||y^{[5]}||_{\text{sup}} \approx 7$ and with $n = 8$, $||y^{[8]}||_{\text{sup}} \approx 13$.

Example 2.2.4. In the last of this series of examples we keep the same ordinary differential equation but choose boundary conditions in such a way that there are infinitely many solutions:

$$y'' = -y, \qquad y(0) = 1, \quad y(2\pi) = 1.$$

A one-parameter family of solutions is $y_c(t) = c\cos t$. As in the previous example, as the number n of iterations is increased $||y^{[n]}||_{\sup}$ becomes increasingly large. For example, with $n = 5$, $||y^{[5]}||_{\sup} \approx 230$ and with $n = 8$, $||y^{[8]}||_{\sup} \approx 235,000$.

In the following examples the right hand side of the differential equation is non-polynomial so we incorporate the Parker-Sochacki method of auxiliary variables in order to insure the Algorithm 2.1.1 is computationally feasible.

Example 2.2.5. Consider the boundary value problem

$$y'' = -\sin y, \qquad y(0) = 1, \quad y(1) = 0. \tag{2.18}$$

As with Examples 2.2.1 and 2.2.2, here the length of the interval $[a,b] = [0,1]$ is much greater than the length on which Theorems 2.1.3 and 2.1.4 guarantee existence of a unique solution to which iterates generated by Algorithm 2.1.1 must converge. The Lipschitz constant of $f(t,y,y') = -\sin y$ is $L = 1$ so the theorems guarantee a result only for times up to $\frac{2}{5}$. In fact the solution is unique and is given by

$$y(t) = k \arccos\left(\frac{2e^{t-1}}{e^{2(t-1)} + 1}\right), \qquad k = \left[\arccos\left(\frac{2e^{-1}}{e^{-2} + 1}\right)\right]^{-1}.$$

In order to effectively apply Algorithm 2.1.1 to (2.18), after introducing the variable $u = y'$ so as to obtain the first order system $y' = u$, $u' = -\sin y$ (the (2.15) in the discussion at the end of the previous section), we must also follow the procedure detailed in the text following (2.15) and illustrated in the discussion surrounding Examples 1.1.3 and 1.1.5.

It is apparent that we must introduce a variable $v = \sin y$, and since the derivative of sine is

cosine, in addition a variable $w = \cos y$. Thus in this instance (2.16) is

$$
\begin{aligned}
y' &= u \\
u' &= -v \\
v' &= uw \\
w' &= -uv
\end{aligned}
$$

with initial conditions

$$
y(0) = 1, \quad u(0) = \gamma, \quad v(0) = \sin 1, \quad w(0) = \cos 1,
$$

a system on $\mathbb{R}^4$ for which the y-component is the solution of the boundary value problem (2.2.5)).

Thus in this instance

$$
a = 0, \quad b = 1, \quad \alpha = 1, \quad \beta = 0,
$$

$$
P_0(y, u, v, w) = -v, \quad P_1(y, u, v, w) = uw, \quad P_2(y, u, v, w) = -uv
$$

so that (??) is

$$
\begin{aligned}
y^{[0]}(t) &\equiv 1 \\
u^{[0]}(t) &\equiv -1 \\
v^{[0]}(t) &\equiv \sin 1 \\
w^{[0]}(t) &\equiv \cos 1 \\
\gamma^{[0]} &= -1
\end{aligned} \tag{2.19a}
$$

and

$$
\begin{pmatrix} y^{[k+1]}(t) \\ u^{[k+1]}(t) \\ v^{[k+1]}(t) \\ w^{[k+1]}(t) \end{pmatrix} = \begin{pmatrix} 1 \\ -1 + \int_0^1 (1-s) v^{[k]}(s)\, ds \\ \sin 1 \\ \cos 1 \end{pmatrix} + \int_0^t \begin{pmatrix} u^{[k]}(s) \\ -v^{[k]}(s) \\ u^{[k]}(s) w^{[k]}(s) \\ -u^{[k]}(s) v^{[k]}(s) \end{pmatrix} ds, \tag{2.19b}
$$

where, as indicated just after (??), we have shifted the update of γ and incorporated it into the update of $u^{[k]}(t)$.

The exact solution $y(t)$, the approximations $y^{[k]}(t)$, and the approximations $\gamma^{[k]}$ to the exact value $\gamma = 1$, with coefficients rounded to five decimal places, are

$$\gamma^{[0]} = 1, \quad \gamma^{[1]} = -0.57926, \quad \gamma^{[2]} = -0.66931, \quad \gamma^{[3]} = -0.68248, \quad \gamma^{[4]} = -0.67738, \quad \gamma^{[5]} = -0.68131$$

and

$$y^{[1]} = 1$$

$$y^{[2]} = 1 - 1.0t$$

$$y^{[3]} = 1 - 0.57926t - 0.42074t^2$$

$$y^{[4]} = 1 - 0.66931t - 0.42074t^2 + 0.09005t^3$$

$$y^{[5]} = 1 - 0.68248t - 0.42074t^2 + 0.05216t^3 + 0.03925t^4 + 0.01180t^5$$

$$y = 1 - 0.74853t - 0.28504t^2 - 0.01997t^3 + 0.03610t^4 + 0.01975t^5$$

A plot of the error in the final approximation $y^{[5]}(t)$ is shown in Figure 2.2.

In the next example we treat a problem in which the function f on the right hand side fails to be Lipschitz in **y** yet Algorithm 2.1.1 nevertheless performs well.

Example 2.2.6. Consider the boundary value problem

$$y'' = -e^{-2y}, \qquad y(0) = 0, \quad y(1.2) = \ln\cos 1.2 \approx -1.015\,123\,283, \tag{2.20}$$

for which the unique solution is $y(t) = \ln\cos t$, yielding $\gamma = 0$.

Introducing the dependent variable $u = y'$ to obtain the equivalent first order system $y' = u$, $u' = e^{-y}$ and the variable $v = e^{-2y}$ to replace the transcendental function with a polynomial we
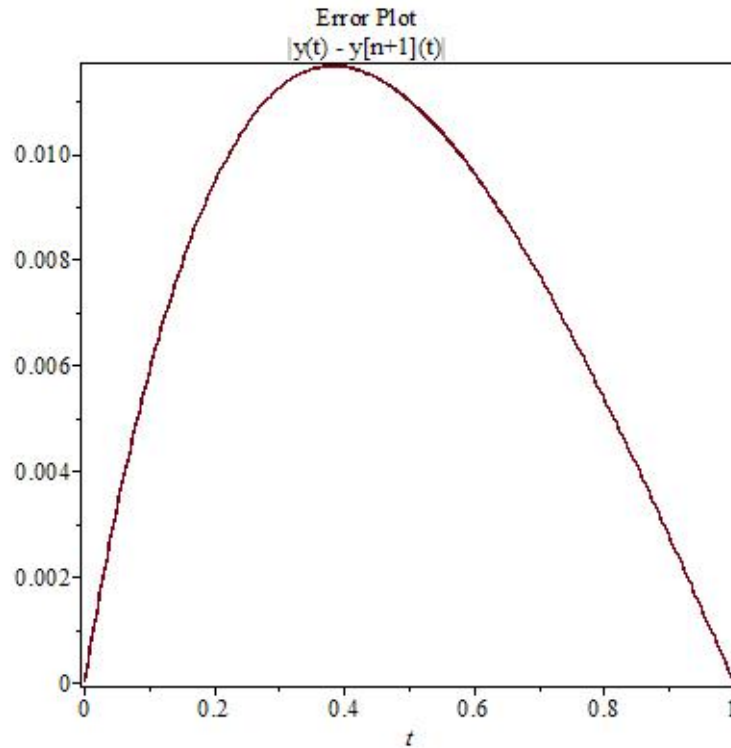
Figure 2.2: Error plot $|y_{\text{exact}}(t) - y^{[5]}(t)|$ for Example 2.2.5.

obtain the expanded system

$$y' = u$$

$$u' = -v$$

$$v' = -2uv$$

with initial conditions

$$y(0) = 0, \quad u(0) = \gamma, \quad v(0) = 1,$$

a system on $\mathbb{R}^3$ for which the y-component is the solution of the boundary value problem (2.20).

Thus in this instance

$$a = 0, \quad b = 1.2, \quad \alpha = 0, \quad \beta = \ln \cos 1.2$$

$$P_0(y, u, v) = -v, \quad P_1(y, u, v) = -2uv$$

so that **(??)** is

$$y^{[0]}(t) \equiv 0$$

$$u^{[0]}(t) \equiv \frac{\ln \cos 1.2}{1.2}$$

$$v^{[0]}(t) \equiv 1$$

$$\gamma^{[0]} = \frac{\ln \cos 1.2}{1.2}$$

(2.21a)

and

$$\begin{pmatrix} y^{[k+1]}(t) \\ u^{[k+1]}(t) \\ v^{[k+1]}(t) \end{pmatrix} = \begin{pmatrix} 0 \\ \frac{1}{1.2}(\beta - \alpha + \int_0^{1.2}(1.2-s)v^{[k]}(s)\,ds) \\ 1 \end{pmatrix} + \int_0^t \begin{pmatrix} u^{[k]}(s) \\ -v^{[k]}(s) \\ -2u^{[k]}(s)v^{[k]}(s) \end{pmatrix} ds, \qquad (2.21b)$$

where, as indicated just after **(??)**, we have shifted the update of γ and incorporated it into the update of $u^{[k]}(t)$. The first eight iterates of γ are:

$$\gamma^{[1]} = -0.24594, \quad \gamma^{[2]} = 0.16011, \quad \gamma^{[3]} = 0.19297, \quad \gamma^{[4]} = 0.04165,$$

$$\gamma^{[5]} = -0.04272, \quad \gamma^{[6]} = -0.04012, \quad \gamma^{[7]} = -0.00923, \quad \gamma^{[8]} = 0.01030,$$

The maximum error in the approximation is $||y_{\text{exact}} - y^{[8]}||_{\sup} \approx 0.0065$.

Example 2.2.7. Consider the boundary value problem

$$y'' = \frac{1}{8}\left(32 + 2t^3 - yy'\right), \qquad y(1) = 17, \quad y(3) = \frac{43}{3} \qquad (2.22)$$

whose right hand side is not autonomous and does not satisfy a global Lipschitz condition. The unique solution is $y(t) = t^3 + \frac{16}{t}$. Using the iterates (2.8) in Algorithm 2.1.1 in order to keep track of the iterates $\gamma^{[k]}$ we have

$$y^{[0]}(t) \equiv 17$$

$$u^{[0]}(t) \equiv -\frac{4}{3}$$

$$\gamma^{[0]} = -\frac{4}{3}$$

(2.23a)

and

$$\begin{pmatrix} y^{[k+1]}(t) \\ u^{[k+1]}(t) \end{pmatrix} = \begin{pmatrix} 17 \\ \gamma^{[k]} \end{pmatrix} + \int_1^t \begin{pmatrix} u^{[k]}(s) \\ 4 + \frac{1}{4}s^3 - y^{[k]}(s)u^{[k]}(s) \end{pmatrix} ds,$$

$$\gamma^{[k+1]} = -\frac{4}{3} - \frac{1}{2}\int_1^3 (3-s)(4 + \frac{1}{4}s^3 - y^{[k]}(s)u^{[k]}(s))\,ds. \qquad (2.23b)$$

After $n = 12$ iterations, $\gamma^{[12]} = -8.443$, whereas the exact value is $\gamma = -\frac{76}{9} \approx -8.444$. As illustrated by the error plot, Figure 2.3 the maximum error in the twelfth approximating function is $||y_{\text{exact}} - y^{[12]}||_{\sup} \approx 0.0012$.
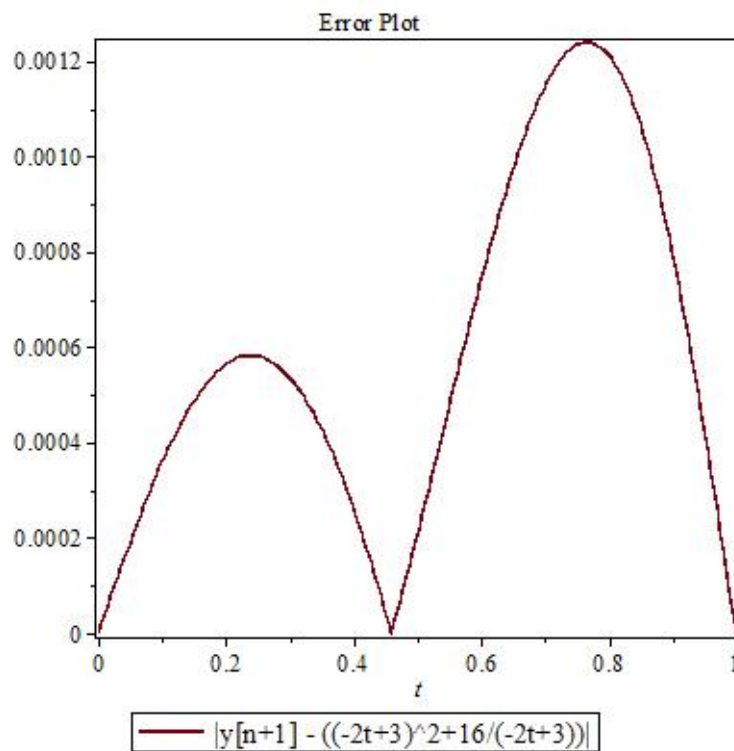

Figure 2.3: Error plot $|y_{\text{exact}}(t) - y^{[12]}(t)|$ for Example 2.2.7.

2.3   Mixed Type Boundary Conditions

The ideas developed at the beginning of Section 2.1 can also be applied to two-point boundary value problems of the form

$$y'' = f(t,y,y'), \qquad y'(a) = \gamma, \quad y(b) = \beta, \qquad (2.24)$$

so that in equation (2.2) the constant γ is now known and $\alpha = y(a)$ is unknown. Thus in (2.6) we evaluate at $t = b$ but now solve for α instead of γ, obtaining in place of (2.7) the expression

$$\alpha = \beta - \gamma(b-a) - \int_a^b (b-s)f(s,y(s),y'(s))\,ds.$$

In the Picard iteration scheme we now successively update an approximation of α starting with some initial value α_0. The iterates are

$$y^{[0]}(t) \equiv \alpha_0$$
$$u^{[0]}(t) \equiv \gamma \tag{2.25a}$$
$$\alpha^{[0]} = \alpha_0$$

and

$$y^{[k+1]}(t) = \alpha^{[k]} + \int_a^t u^{[k]}(s)\,ds$$
$$u^{[k+1]}(t) = \gamma + \int_a^t f(s,y^{[k]}(s),u^{[k]}(s))\,ds \tag{2.25b}$$
$$\alpha^{[k+1]} = \beta - \gamma(b-a) - \int_a^b (b-s)f(s,y^{[k]}(s),u^{[k]}(s))\,ds.$$

We obtain in analogy with Algorithm 2.1.1 the following algorithm for approximating solutions of (2.24), in which we have changed the update of α in (2.25) to incorporate it into the update of y.

Algorithm 2.3.1. To approximate the solution of the boundary value problem

$$y'' = f(t,y,y'), \qquad y'(a) = \gamma, \quad y(b) = \beta \tag{2.26}$$

select an initial value α_0 and iteratively compute the sequence of functions on $[a,b]$

$$y^{[0]}(t) \equiv \alpha_0$$
$$u^{[0]}(t) \equiv \gamma \tag{2.27a}$$

and

$$y^{[k+1]}(t) = \beta - \gamma(b-a) - \int_a^b (b-s)f(s,y^{[k]}(s),u^{[k]}(s))\,ds + \int_0^t u^{[k]}(s)\,ds$$
$$u^{[k+1](t)} = \gamma + \int_a^t f(s,y^{[k]}(s),u^{[k]}(s))\,ds. \tag{2.27b}$$

An exact analogue of Theorem 2.1.3 but for the boundary conditions in (2.26) can be proved exactly as Theorem 2.1.3 was proved, by means of a contraction mapping argument. Thus exactly

as before the second statement in the Contraction Mapping Theorem (Theorem 1.3.5) and the fact that repeated composition of the mapping $\mathcal{T}$ in the proof of these theorems generates Picard iterates guarantees that the iterates defined by (2.27) will converge to the unique solution of (2.26) that the analogue of Theorem 2.1.3 guarantees to exist. Thus we have the following result analogous to Theorem 2.1.4.

Theorem 2.3.2. *Let $f : [a,b] \times \mathbb{R}^2 \to \mathbb{R} : (t,y,u) \mapsto f(t,y,u)$ be Lipschitz in $\mathbf{y} = (y,u)$ with Lipschitz constant L with respect to absolute value on $\mathbb{R}$ and the sum norm on $\mathbb{R}^2$. If $0 < b - a < (1 + \frac{3}{2}L)^{-1}$ then for any β, $\gamma \in \mathbb{R}$ the iterates generated by Algorithm 2.3.1 converge to the unique solution of the boundary value problem*

$$y'' = f(t,y,y'), \qquad y'(a) = \gamma, \quad y(b) = \beta$$

guaranteed by the analogue of Theorem 2.1.3 to exist.

Example 2.3.3. Consider the boundary value problem

$$y'' = -y'e^{-y}, \qquad y'(0) = 1, \quad y(1) = \ln 2 \tag{2.28}$$

The exact solution is $y(t) = \ln(1+t)$, for which $y(0) = \ln 1 = 0$. Introducing the dependent variable $u = y'$ as always to obtain the equivalent first order system $y' = u$, $u' = e^{-y}$ and the variable $v = e^{-y}$ to replace the transcendental function with a polynomial we obtain the expanded system

$$y' = u$$
$$u' = -uv$$
$$v' = -uv$$

with initial conditions

$$y(0) = \alpha, \quad u(0) = \gamma, \quad v(0) = e^{-\alpha},$$

a system on $\mathbb{R}^3$ for which the y-component is the solution of the boundary value problem (2.28).

Thus in this instance

$$a = 0, \quad b = 1, \quad \beta = \ln 2, \quad \gamma = 1$$

$$g_0(y, u, v) = -uv, \quad g_1(y, u, v) = -uv$$

so that, making the initial choice $\alpha_0 = 1$, the analogue of (??) is

$$y^{[0]}(t) \equiv \alpha_0$$
$$u^{[0]}(t) \equiv 1 \qquad\qquad\qquad (2.29a)$$
$$v^{[0]}(t) \equiv e^{-\alpha_0}$$

and

$$\begin{pmatrix} y^{[k+1]}(t) \\ u^{[k+1]}(t) \\ v^{[k+1]}(t) \end{pmatrix} = \begin{pmatrix} \ln 2 - 1 + \int_0^1 (1-s) u^{[k]}(s) v^{[k]}(s)\, ds \\ 1 \\ v^{[0]} \end{pmatrix} + \int_0^t \begin{pmatrix} u^{[k]}(s) \\ -u^{[k]}(s) v^{[k]}(s) \\ -u^{[k]}(s) v^{[k]}(s) \end{pmatrix} ds, \qquad (2.29b)$$

where we have shifted the update of α and incorporated it into the update of $y^{[k]}(t)$. We list the first eight values of α to show the rate of convergence to the exact value 0.

$$\alpha^{[1]} = -0.00774, \quad \alpha^{[2]} = 0.11814, \quad \alpha^{[3]} = 0.07415, \quad \alpha^{[4]} = 0.08549,$$

$$\alpha^{[5]} = 0.08288, \quad \alpha^{[6]} = 0.08339, \quad \alpha^{[7]} = 0.08330, \quad \alpha^{[8]} = 0.08331$$

2.4   Mixed Type Boundary Conditions: A Second Approach

In actual applications Algorithm 2.3.1 converges relatively slowly because of the update at every step of the estimate of the initial *value* $y(a)$ rather than of the initial *slope* $y'(a)$. A different approach that works better in practice with a problem of the type

$$y'' = f(t, y, y'), \qquad y'(a) = \gamma, \quad y(b) = \beta, \qquad\qquad (2.30)$$

is to use the relation

$$y'(b) = y'(a) + \int_a^b f(s, y(s), y'(s))\, ds \qquad\qquad (2.31)$$

between the derivatives at the endpoints to work from the right endpoint of the interval $[a,b]$, at which the value of the solution y is known but the derivative y' unknown. That is, assuming that (2.30) has a unique solution and letting the value of its derivative at $t = b$ be denoted δ, it is also the unique solution of the initial value problem

$$y'' = f(t,y,y'), \qquad y(b) = \beta, \quad y'(b) = \delta. \tag{2.32}$$

We introduce the new dependent variable $u = y'$ to obtain the equivalent system

$$y' = u$$
$$u' = f(t,y,u)$$

with initial conditions $y(b) = \beta$ and $y'(b) = \delta$ and apply Picard iteration based at $t = b$, using (2.31) with $y'(a) = \gamma$ to update the approximation of $y'(b)$ are each step. Choosing a convenient initial estimate δ_0 of δ, the successive approximations of the solution of (2.32), hence of (2.30), are given by

$$y^{[0]}(t) \equiv \beta$$
$$u^{[0]}(t) \equiv \delta_0 \tag{2.33a}$$
$$\delta^{[0]} = \delta_0$$

and

$$y^{[k+1]}(t) = \beta + \int_b^t u^{[k]}(s)\,ds$$
$$u^{[k+1]}(t) = \delta^{[k]} + \int_b^t f(s, y^{[k]}(s), u^{[k]}(s))\,ds \tag{2.33b}$$
$$\delta^{[k+1]} = \gamma + \int_a^b f(s, y^{[k]}(s), u^{[k]}(s))\,ds.$$

This gives the following alternative algorithm for approximating solutions of (2.30).

Algorithm 2.4.1. To approximate the solution of the boundary value problem

$$y'' = f(t,y,y'), \qquad y'(a) = \gamma, \quad y(b) = \beta \tag{2.34}$$

select an initial value δ_0 and iteratively compute the sequence of functions on $[a,b]$ given by (2.33).

By the theorem mentioned in Section 2.3 that is the analogue of Theorem 2.1.3 we obtain the analogue of Theorems 2.1.4 and 2.3.2 that guarantees convergence of the iterates to the unique solution of (2.34).

Example 2.4.2. Reconsider the boundary value problem (2.28) of Example 2.3.3,

$$y'' = -y'e^{-y}, \qquad y'(0) = 1, \quad y(1) = \ln 2 \tag{2.35}$$

with exact solution $y(t) = \ln(1+t)$. The relation (2.31) is

$$y'(1) = 1 - \int_0^1 y'(s)e^{-y(s)}\, ds. \tag{2.36}$$

Exactly as in Example 2.3.3 we set $u = y'$ and $v = e^{-y}$ to obtain the expanded system

$$y' = u$$

$$u' = -uv$$

$$v' = -uv$$

but now with initial conditions

$$y(1) = \ln 2, \quad u(1) = \delta, \quad v(1) = \tfrac{1}{2},$$

a system on $\mathbb{R}^3$ for which the y-component is the solution of the boundary value problem (2.35). Making the initial choice $\delta_0 = 1$ (assuming that the slope from one endpoint to the next will not change too drastically) and using (2.36) for the update of δ, (2.33) as applied to the expanded system with the auxiliary variable v is

$$y^{[0]}(t) \equiv \ln 2$$
$$u^{[0]}(t) \equiv 1$$
$$v^{[0]}(t) \equiv \tfrac{1}{2} \tag{2.37a}$$
$$\delta^{[0]} = 1$$

and

$$
\begin{pmatrix} y^{[k+1]}(t) \\ u^{[k+1]}(t) \\ v^{[k+1]}(t) \end{pmatrix} = \begin{pmatrix} \ln 2 \\ \delta^{[k]} \\ \frac{1}{2} \end{pmatrix} + \int_0^t \begin{pmatrix} u^{[k]}(s) \\ -u^{[k]}(s)v^{[k]}(s) \\ -u^{[k]}(s)v^{[k]}(s) \end{pmatrix} ds,
$$

$$
\delta^{[k+1]} = 1 - \int_0^1 u^{[k]}(s)v^{[k]}(s)\, ds.
$$

(2.37b)

The exact value of $\delta = y'(1)$ is $\frac{1}{2}$. The first eight values of $\delta^{[k]}$ are:

$$\delta^{[0]} = -0.50000, \quad \delta^{[1]} = 0.41667, \quad \delta^{[2]} = 0.50074,$$

$$\delta^{[3]} = 0.51592, \quad \delta^{[4]} = 0.49987, \quad \delta^{[5]} = 0.49690,$$

$$\delta^{[6]} = 0.50003, \quad \delta^{[7]} = 0.50060, \quad \delta^{[8]} = 0.50000$$

CHAPTER 3: COMPARISON OF METHODS

We will compare the method presented in Chapters 1 and 2 against the methods are commonly used in the literature. These are the Shooting Method using Fourth Order Runge-Kutta Approximation, the Finite Difference Method, the Adomian Decomposition Method, and the Power Series Method.

To show the efficacy and applicability of our method, we will compare all these methods on a linear boundary value problem and then a non-linear boundary value problem.

3.1   Linear Cases

A linear second order two-point boundary value problem, namely that

$$y'' = f(t, y, y')$$

$$y(a) = \alpha$$

$$y(b) = \beta$$

is *linear* if $f(t, y, u) = q(t)y + p(t)u + r(t)$. Theorem (1.2.2) guarantees the unique solution for the second-order linear boundary value problem.

$$y'' = -y$$

$$y(0) = 1 \tag{3.1}$$

$$y(1) = 0$$

We will compare methods applied to (3.1), whose exact solution is (3.2),

$$y(t) = \cos t - \frac{\cos 1}{\sin 1} \sin t \tag{3.2}$$

3.1.1 Runge Kutta with Shooting Method

The method is based on the decomposition (1.13) of the solution of (1.12) given in Theorem 1.2.2. The unknown solution of the boundary value problem (1.12) can be expressed as a linear combination of the solutions of the initial value problems (1.14)) and (1.15), each of which, after it has been expressed as a first order system, can be approximated by a numerical scheme, in this case the fourth order Runge-Kutta method, using the approximate values at the right endpoint to approximate the second coefficient $(\beta - y_1(b))/y_2(b)$.

A key point is that there are parameter choices to be made in applying this method. In particular, the number n of partition points of the interval $[a,b]$ must be selected in advance. Another important point that should be made is that the output of this method is not a function, as with our algorithm, but a sequence of $n+1$ points that approximate the values of the unknown solutions at $n+1$ t-values. We are able to compute the error only at these $n+1$ points and then have linearly interpolated between them.

Applying this method requires approximating the solutions to the initial value problems

$$
\begin{aligned}
y'' &= -y \\
y(0) &= 1 \\
y'(0) &= 0
\end{aligned}
\tag{3.3}
$$

and

$$
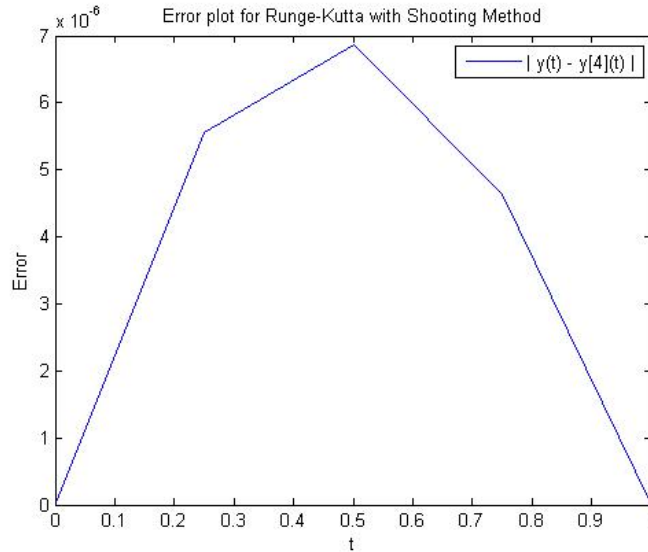\begin{aligned}
y'' &= -y \\
y(0) &= 0 \\
y'(0) &= 1
\end{aligned}
\tag{3.4}
$$

If $y_1(t)$ denotes the solutions to equation (3.3) and $y_2(t)$ denotes the solutions to equation (3.4), then

$$
y(t) = y_1(t) + \frac{\beta - y_1(1)}{y_2(1)} y_2(t)
$$

is the unique solution to the boundary value problem (3.1).

A plot of the error in the final approximation is shown in below

3.1.2 Finite difference method

This method reduced a differential equation to an algebraic equation by replacing the derivatives by their finite difference approximations. Once the number n of subintervals into which $[a,b]$ is partitioned the approximation technique leads to an $(n-1) \times (n-1)$ system of linear equations whose solution gives as output a sequence of $n+1$ points that approximate the values of the unknown solutions at $n+1$ t-values. The parameter choice that we need to made in advance is the number n of partition points of the interval $[a,b]$. Then we can set the mesh size $h = \frac{b-a}{n}$ and then nodes $t_i = a + ih$. Similar to the "Runge-Kutta" the output of this method is not a function but a sequence of $n+1$ points that approximate the values of the unknown solutions at $n+1$ t-values. We are able to compute the error only at these $n+1$ points and then have linearly interpolated between them.

$$y'' = f(t,y,y'), \quad a \leq t \leq b$$
$$y(a) = \alpha, \quad y(b) = \beta, \qquad \text{Bounadary Condition}$$

(3.5)

To applying the method to the equation (3.1)first subdivide the interval $[0,1]$ into n equal subintervals. The nodes of this subdivision are $t_i = ih$, $i = 0, \cdots, n$ where $h = \frac{b-a}{n}$. We define $y_i = y(t_i)$ and from the boundary condition we have $y_0 = 1$ and $y_n = 0$. From the central difference formula we have

$$y'' \approx \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2}$$

hence the equation (3.1) becomes

$$y_{i+1} - 2y_i + y_{i-1} + h^2 y_i = 0, \qquad i = 1, \cdots, n-1$$

from here obtaining y_is are straightforward and this sequence of partition points of the interval $[0,1]$ is the solution to the equation (3.1. For problem (3.1) we set $n = 4$ partition points. A plot of the error in the final approximation is shown in below
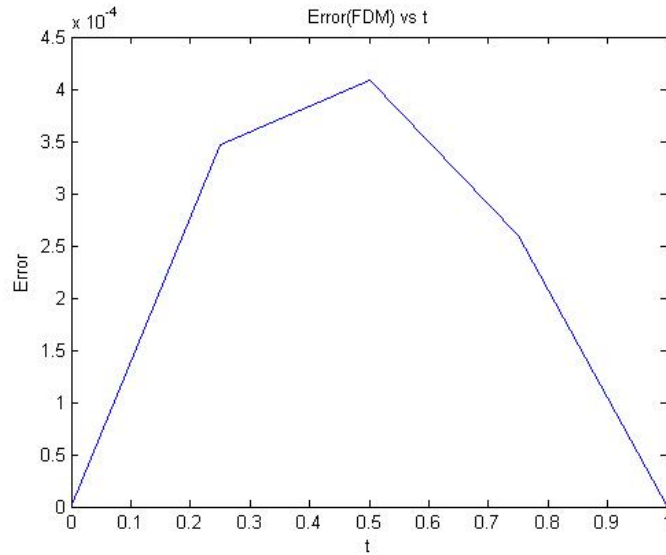
3.1.3 Adomian

As we may have noticed, most realistic system of ordinary differential equations do not have analytic solutions so that a numerical technique must be used. But in the case of using the Adomians decomposition method is that it can provide analytical approximation to the problems. This method decompose the differential equations into Three operators . *Invertible* operator, *Linear* operator and *Non-Linear* operator. The invertible operator is the the highest order of differential equation. The output of this method is a function and we will see the outputs of Power Series and Our new methods are also a function.

Consider

$$\frac{d^2}{dt^2} y = f(t) + R(y) + N(y), \quad 0 \le t \le 1$$
$$y(0) = \alpha \tag{3.6}$$
$$y(h) = \beta$$

Where $L = \frac{d^2}{dt^2}$, is invertible operator,it can be written as $L^{-1}(\cdot) = \int_0^t \int_0^s \cdot dx ds$, N is nonlinear operator and R is stand for remaining part. The solution will be in form of $y = \sum_{k=0}^{\infty} y_n$, The way the method technically works is we compute $\phi_n(t) = \sum_{k=0}^{n} y_n$ and then $y = \lim_{n \to \infty} \phi_n$. If the limit can be taken then a analytic solution can be obtained. The number n of iteration must be selected in advance.

Applying the inverse linear operator $L^{-1}(\cdot) = \int_0^t \int_0^s \cdot dx ds$ to both sides of equation (3.1) and using boundary value at $t = 0$ yields

$$y(t) = 1 + \gamma t - L^{-1}y(t)$$

where $\gamma = y'(0)$ is an unknown parameter.

By the standard recursion ascheme

$$y_0 = 1 + \gamma t$$

$$y_n = -L^{-1}y_{n-1}, \qquad n = 1, 2, \cdots$$

We calculate the solution component as

$$y_1 = -\frac{1}{2}t^2 + \frac{1}{6}\gamma t^3$$

$$y_2 = -\frac{1}{24}t^4 + \frac{1}{120}\gamma t^5$$

$$\cdots$$

And the approximation solution is given by

$$\phi_n(t) = \sum_{k=0}^{n-1} y_n(t)$$

Therefore the approximate solutions are

$$\phi_1 = 1 + \gamma t$$

$$\phi_2 = 1 + \gamma t - \frac{1}{2}t^2 + \frac{1}{6}\gamma t^3$$

$$\phi_3 = 1 + \gamma t - \frac{1}{2}t^2 + \frac{1}{6}\gamma t^3 - \frac{1}{24}t^4 + \frac{1}{120}\gamma t^5$$

$$\ldots$$

By matching the $\phi_n(t)$ at $t = 1$, we can then determine the value of γ as a convergent sequence of approximate values. If we match $\phi_3(t)$ at $t = 1$, then we need to solve

$$1 + \gamma t - \frac{1}{2}t^2 + \frac{1}{6}\gamma t^3 - \frac{1}{24}t^4 + \frac{1}{120}\gamma t^5 = 0$$

we obtain $\gamma = -0.6435643568$, and thus the approximate solution is

$$\phi_3(t) = 1 - 0.6435643568\,t - \frac{1}{2}t^2 - \frac{0.6435643568}{6}t^3 - \frac{1}{24}t^4 - \frac{0.6435643568}{120}t^5$$

The exact value of γ is $y'(0) = -0.6420926160$. The way Adomian explained in its standard form, required it to be solved symbolically. However, if we would like to use MATLAB to solve it then the common procedure is to solve it by linear shooting method which makes it tequivalent to Linear Power Series method. A plot of the error in the final approximation $y^{[5]}(t)$ is shown in below
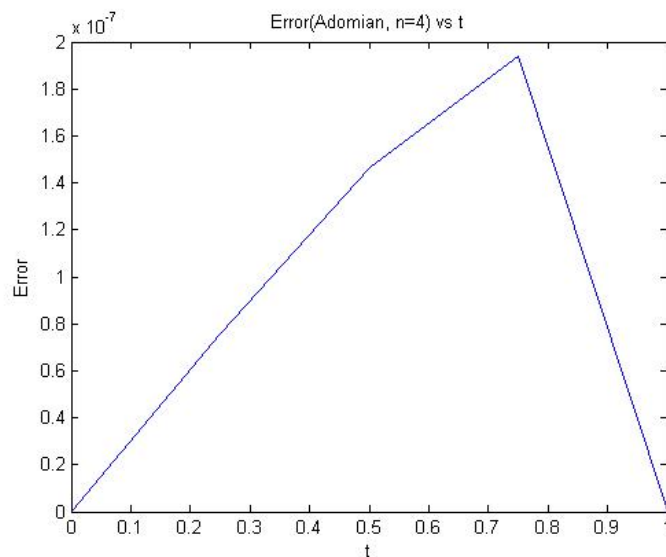
3.1.4 Power Series Method

The method is feasible because it is based on the decomposition (1.13) of the solution of (1.12) given in Theorem 1.2.2. We assume the solution to the differential equations (1.14)) and (1.15) can be represented as a power series of order n. We can recursively find coefficients in the power series because now our two conditions are at the same point. The unknown solution is expanded in powers of $t - a$. The output of this method is a function. To apply the method we need to the number of iterations n in advance.

Applying this method to the equation (3.1) requires approximating the solutions to the initial value problems (3.3) and (3.4). If $y_1(t) = \sum_{i=0}^{n} a_i t^i$ denotes the solutions to equation (3.3) by substituting it into equations (3.3) we obtain

$$a_n = -\frac{a_{n-2}}{n(n-1)}$$

and from the initial value is easy to see that $a_0 = 1$ and $a_1 = 0$ therefore the approximate solution $y_1(t)$ with $n = 4$ is

$$y_1(t) = 1 - \frac{1}{2}t^2 + \frac{1}{24}t^4 \tag{3.7}$$

If $y_2(t) = \sum_{i=0}^{n} b_i t^i$ denotes the solutions to equation (3.4) similarly we obtain

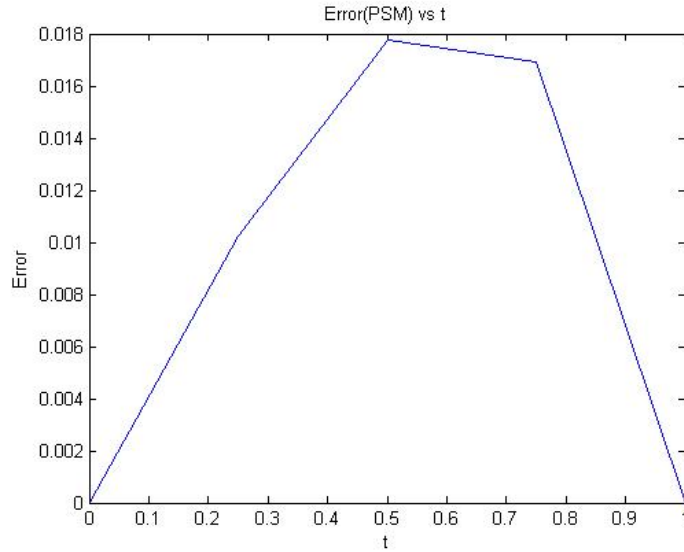$$y_2(t) = t - \frac{1}{6}t^3 \tag{3.8}$$

then by using

$$y(t) = y_1(t) + \frac{\beta - y_1(1)}{y_2(1)} y_2(t)$$

we obtain the the approximate solution

$$y(t) = 1 - 0.65000t - 0.50000t^2 + 0.10833t^3 + 0.041667t^4$$

to the solution of (3.1).

A plot of the error in the final approximation is shown in below

Error(PSM) vs t

3.1.5 Our New Method

Now we are going to implement our new method to solve boundary value problem (3.1). We have already explained the process on Chapter (2.2). Since the method is based on picard's iteration then knowledge of initial guess is necessary.

We have introduced the $\gamma = y'(0)$ as an unknown parameter. The output of this method is a function.

The exact solution $y(t)$, the approximations $y^{[5]}(t)$, with coefficients rounded to five decimal places, are

$$y = 1 - 0.64210t - 0.50000t^2 + .10702t^3 + 0.04167t^4 + 0.00534t^5$$

$$y^{[5]} = 1 - 0.64444t - 0.50000t^2 + 0.11111t^3 + 0.04167t^4 + 0.00433t^5$$

and the exact value $\gamma = -0.64210$ can be approximated to $\gamma^{[4]} = -0.64444$. A plot of the error in the final approximation $y^{[5]}(t)$ is shown in the following Figure.

Error Plot
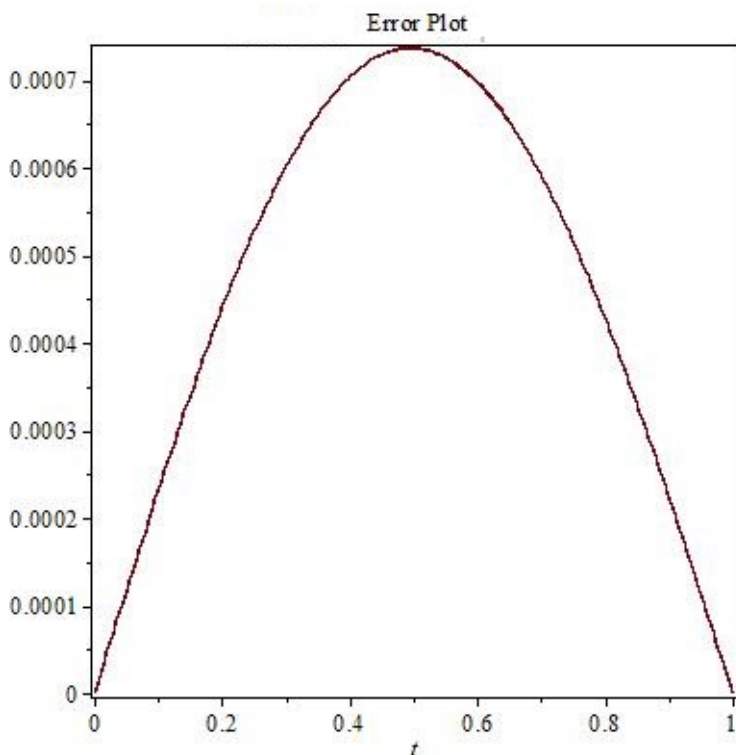
3.1.6 A Comparison of Methods

The following table compares the results of the five methods on problem (3.1). It gives the the approximation to the exact solution at the five partition points common to the Shooting and Finite Difference methods.

t	$Runge-Kutta$	$Finite-Difference$	$Power-Series$	$Adomian$	$New-Method$	$Exact$
0.000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000
0.250	0.810051	0.810403	0.820313	0.810056	0.809530	0.810056
0.500	0.569740	0.570156	0.587500	0.569747	0.569010	0.569747
0.750	0.294009	0.294274	0.310938	0.294013	0.2934977	0.294014
1.000	0.000000	0.000000	0.000000	-0.000000	0.0000000	0.000000

Our method, like the power series method and Adomian, effectively approximates the solution at every point of the interval $[a,b]$, whereas the other methods give only a sequence of approximation points. Thus no interpolation is needed for our method to give approximate values anywhere in the interval.

To compute elapsed time for each method we have written them y MATLAB and use the "tic" "toc"

command. The calculated elapsed time for each methods is given as follow,

$Runge-Kutta$	$Finite-Difference$	$Power-Series$	$New-Methodt$
10 $steps$	10 $steps$	$n=10$	$n=10$
0.003175	0.013968	0.004003	0.002682

3.2   Non-Linear Case

In this section we apply the same five approximation methods introduced in previous section to the nonlinear boundary value problem

$$y'' = -\sin y, \quad y(0) = 1, \quad y(1) = 0, \tag{3.9}$$

whose exact solution is

$$y(t) = k \arccos\left(\frac{2e^{t-1}}{e^{2(t-1)}+1}\right), \qquad k = \left[\arccos\left(\frac{2e^{-1}}{e^{-2}+1}\right)\right]^{-1}.$$

3.2.1   Runge Kutta with Shooting Method)

The technique for the nonlinear second order boundary value problem

$$y'' = f(t,y,y'), \quad y(a) = \alpha, \quad y(b) = \beta, \tag{3.10}$$

is similar to the linear case, except that the solution to a nonlinear problem can not be simply expressed as a linear combination of the solution to two initial value problems. The idea is to write the boundary value problem (3.10) as a sequence of initial value problems (3.11) and begin the solution at one end of the boundary value problem, and shoot to the other end with an initial value solver until the boundary condition at the other end converges to its correct value. The solutions of the initial value problems can be approximated by a numerical scheme, in this case the fourth order Runge-Kutta method.

$$y'' = f(t,y,y'), \quad y(a) = \alpha, \quad y'(a) = \gamma, \tag{3.11}$$

The method is an iteration procedure based on a guess for an unknown initial condition, which in this case is $y'(a) = \gamma$. Therefore the choices to be made in applying this method are the number n of partition points of the interval $[a,b]$ must be selected in advance and initial guess for γ. We start with a parameter γ_0 that determines the initial elevation and shoot to the other end from the point (a,α) and if the object is not landed sufficiently close to β then we correct our approximation. To correct the initial guess and generate a new

approximation we use Newton's method to generate γ_k, which the method is in form of

$$\gamma_k = \gamma_{k-1} - \frac{y(b,\gamma_{k-1}) - \beta}{\frac{dy}{dt}(b,\gamma_{k-1})}$$

where $y(b,\gamma_j)$ is the solution to initial value problem (3.12) at $t = b$.

$$y'' = f(t,y,y'), \quad y(a) = \alpha, \quad y'(a) = \gamma_j, \tag{3.12}$$

As the linear case the output of this method is not a function but a sequence of $n+1$ points that approximate the values of the unknown solutions at $n+1$ t-values and then have linearly interpolated between them. We applied the method to solve problem (3.9) for $n = 7$ and initial guess $\gamma_0 = \frac{\beta - \alpha}{b - a}$.

A plot of the error in the final approximation $y^{[7]}(t)$ is shown in below
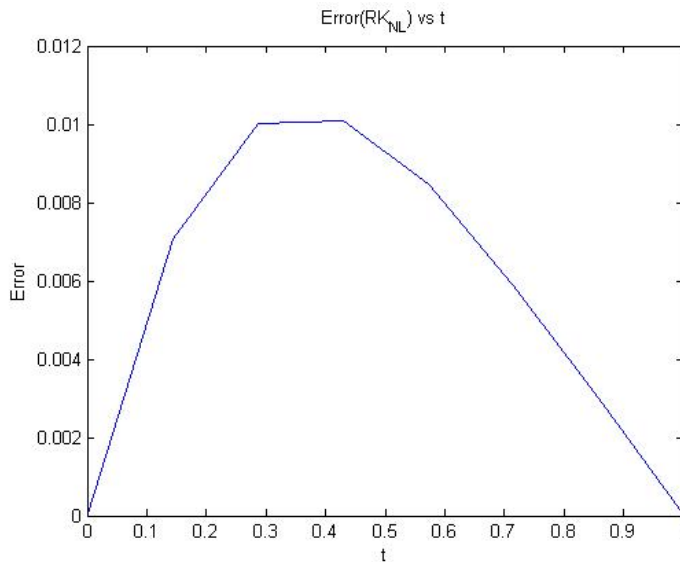
3.2.2 Finite Difference Method

The finite difference approximations for nonlinear case is similar to linear case. Except in this case, the approximation technique leads to an $(n-1) \times (n-1)$ system of *non-linear* equations whose solution gives as output a sequence of $n+1$ points that approximate the values of the unknown solutions at $n+1$ t-values, where n is the number of subintervals into which $[a,b]$ is partitioned. Similar to linear case, equation (3.1) can be written as

$$y_{i+1} - 2y_i + y_{i-1} + h^2 \sin(y_i) = 0, \qquad i = 1, \cdots, n-1$$

because of the term $\sin(y_i)$ then we need to solve a system of nonlinear equation which can be solved and approximated by Newton's method. A plot of the error in the final approximation is shown in below

Error(FDM$_{NL}$) vs t
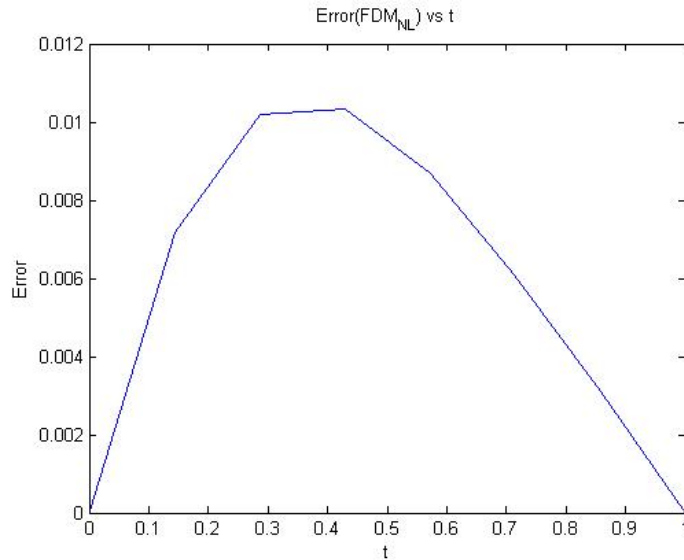
3.2.3 Adomian

Applying the inverse linear operator $L^{-1}(\cdot) = \int_0^t \int_0^s \cdot \, dx\, ds$ to both sides of equation (3.9) and using boundary value at $t = 0$ yields

$$y(t) = 1 + \gamma t - L^{-1} \sin y(t)$$

where $\gamma = y'(0)$ is an unknown parameter.

Next we decompose the solution $y(t)$ and nonlinearity $\sin y(t)$.

$$y(t) = \sum_{n=0}^{\infty} y_n(t)$$

$$\sin y(t) = \sum_{n=0}^{\infty} A_n$$

where the A_n are the Adomian polynomials for the nonlinearity $Ny(t) = -\sin y(t)$ and can be computed by

$$A_n = \frac{1}{n!} \left[\frac{d^n}{d\lambda^n} N \left(\sum_{i=0}^{n} \lambda^i y_i \right) \right]_{\lambda=0}$$

Hence

$$A_0 = -\sin y_0$$

$$A_1 = -y_1 \cos y_0$$

$$A_2 = \frac{1}{2}y_1^2 \sin y_0 - y_2 \cos y_0 \qquad \cdots$$

By the standard recursion ascheme

$$y_0 = 1 + \gamma t$$

$$y_n = L^{-1}A_{n-1}, \qquad n = 1, 2, \cdots$$

We calculate the solution component as

$$y_1 = -\frac{1}{2}\frac{\sin 1 + \cos 1\, t\gamma - sin(1 + \gamma t)}{\gamma^2}$$

$$y_2 = -\frac{1}{4}\frac{5\sin 1\cos 1 + 4\sin^2 1\, t\gamma + 6\cos^2 1\, t\gamma + 4\sin 1\cos(1 + \gamma t)}{\gamma^4}$$

$$+ \frac{-8\cos 1\sin(1 + \gamma t) + 4\cos 1\cos(1 + \gamma t)\gamma t - \cos(1 + \gamma t) - \gamma t}{\gamma^4}$$

$$\cdots$$

And the approximation solution is given by

$$\phi_n(t) = \sum_{k=0}^{n-1} y_n(t)$$

Therefore the approximate solutions are

$$\phi_1 = \sin(1+\gamma t)$$

$$\phi_2 = \sin(1+\gamma t) - \frac{1}{2}\frac{\sin 1 + \cos 1 \,(t\gamma) - sin(1+\gamma t)}{\gamma^2}$$

$$\phi_3 = \sin(1+\gamma t) - \frac{1}{2}\frac{\sin 1 + \cos 1 \,(t\gamma) - sin(1+\gamma t)}{\gamma^2}$$

$$- \frac{1}{4}\frac{5\sin 1 \cos 1 + 4\sin^2 1 \,(t\gamma) + 6\cos^2 1 \,(t\gamma) + 4\sin 1 \cos(1+\gamma t)}{\gamma^4}$$

$$+ \frac{-8\cos 1 \sin(1+\gamma t) + 4\cos 1 \cos(1+\gamma t)\gamma t - \cos(1+\gamma t) - \gamma t}{\gamma^4}$$

$$\dots$$

By matching the $\phi_n(t)$ at $t = 1$, we can then determine the value of γ as a convergent sequence of approximate values. If we match $\phi_2(t)$ at $t = 1$, then we need to solve

$$\sin(1+\gamma) - \frac{1}{2}\frac{\sin 1 + \cos 1 \,(\gamma) - sin(1+\gamma)}{\gamma^2} = 0$$

Therefore, to solve for γ one has to approximate γ using a solver which makes Adomian computationally challenging.

3.2.4   Power Series Method

The general approach to solve non-linear case for this method is same as the linear case. However, because the right hand sides of the first order differential equation system of two point boundary value problems are polynomials, therefore the only difficulty we might encounter in compare to linear case is, in non-linear case we need to apply "Cauchy product" a few times. Here we will solve problem (3.9) by Power Series method. The point that we need to make is in Power Series Method and Our New method we do not need to compute the value of $f(t,y,y')$ instead we will compute the value of a sequence of polynomials which these two methods compotationally friendly. It is apparent that we must introduce a variable $v = \sin y$, and since the derivative of sine is cosine, in addition a variable $w = \cos y$. Thus

$$y' = u$$
$$u' = -v$$
$$v' = uw$$
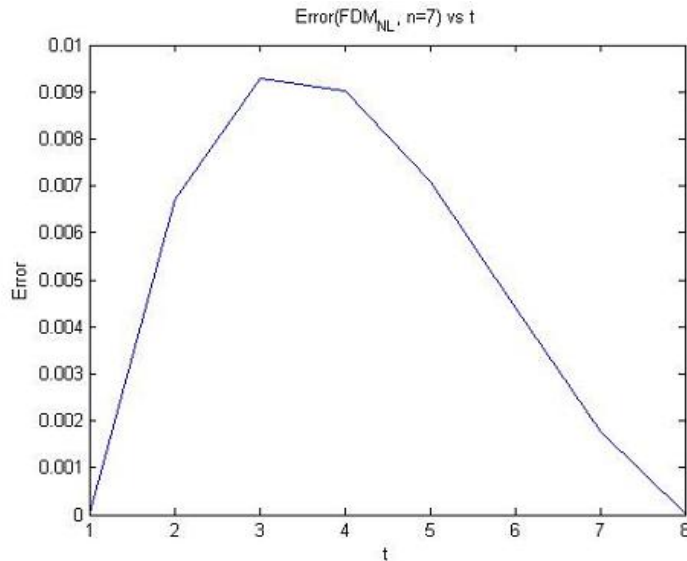$$w' = -uv$$

with initial conditions

$$y(0) = 1, \quad u(0) = \gamma, \quad v(0) = \sin 1, \quad w(0) = \cos 1,$$

We assume that we can write the solutions to the above system as a power series and then try to determine what the coefficients need to be.

After 4 iterations, we get

$$y = 1 - t + 0.42074 t^2 - 0.09005 t^3 - 0.09005 t^4 - 0.01612 t^5 + 0.01977 t^6 - 0.00792 t^7$$

A plot of the error in the final approximation is shown in below

Error(FDM_{NL}, n=7) vs t

3.2.5   Our New Method

Our method take cares of the non-linear case exactly like the linear case, the only extra work that we have to do here is we might need to introduce a few more auxiliary variables than in linear case. The problem (3.9) has been solved for $n = 7$. Chapter2 example (refNex1) explains how to solve the problem in great details. The exact solution $y(t)$, and the approximations $y^{[5]}(t)$, with coefficients rounded to five decimal places for the first 5 terms are

$$y^{[5]} = 1 - 0.68115 t - 0.42074 t^2 + 0.06135 t^3 + 0.03512 t^4 + 0.00894 t^5$$

$$y = 1 - 0.74853 t - 0.28504 t^2 - 0.01997 t^3 + 0.03610 t^4 + 0.01975 t^5$$

A plot of the error in the final approximation $y^{[8]}(t)$ is shown in below

Error Plot
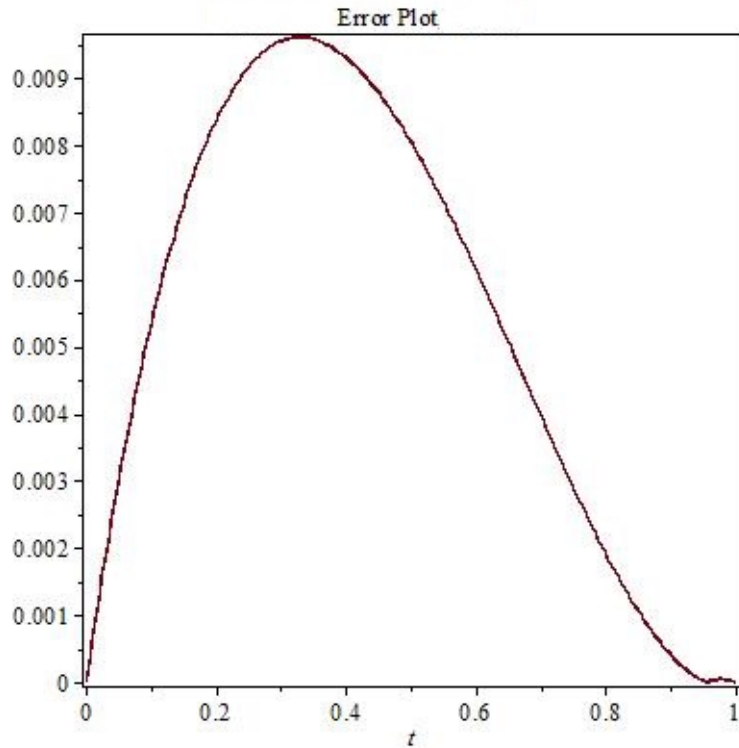
3.2.6 Methods Comparison

A quick comparison between methods is given in the following tables

t	$Runge-Kutta$	$Finite-Difference$	$Power-Series$	$New-Method$	$Exact$
0.000	1.000000	1.000000	1.000000	1.000000	1.000000
0.143	0.894286	0.894394	0.893919	0.894254	0.887208
0.286	0.772692	0.772874	0.771963	0.772630	0.762679
0.429	0.636889	0.637104	0.635814	0.636797	0.626784
0.571	0.488986	0.489194	0.487617	0.488868	0.480525
0.714	0.331531	0.331695	0.330024	0.331399	0.325612
0.857	0.167457	0.167549	0.166228	0.167350	0.164448
1.000	-0.000004	0.000000	-0.000006	0.000000	0.000000

The calculated elapsed time for each methods is given as follow,

$Runge - Kutta$	$Finite - Difference$	$Power - Series$	$New - Methodt$
$10 \, steps$	$10 \, steps$	$n = 10$	$n = 10$
0.032214	0.016092	0.007138	0.005649

CHAPTER 4: VOLTERRA INTEGRAL EQUATION

A Volterra integral equation of the second kind is an equation of the form

$$y(t) = \varphi(t) + \int_a^t K(t,s)f(s,y(s))\,ds \tag{4.1}$$

where φ, K, and f are known functions of suitable regularity and y is an unknown function. Such equations lend themselves to solution by successive approximation using Picard iteration, although in the case that the known functions are not polynomials the process can break down when quadratures that cannot be performed in closed form arise. Equation (4.1) can be generalized to

$$y(t) = \varphi(t) + \int_a^t K(t,s,y(s))\,ds. \tag{4.2}$$

If the "kernel" K is independent of the first variable t then the integral equation is equivalent to an initial value problem. Indeed, in precisely the reverse of the process used in Chapter 2, we have that

$$y(t) = \varphi(t) + \int_a^t K(s,y(s))\,ds$$

is equivalent to

$$y'(t) = \varphi'(t) + K(t,y(t)), \quad y(a) = \varphi(a).$$

In this chapter we provide a method for introducing auxiliary variables in (4.2) in the case that the t-dependence in K factors out, so that $K(t,s,z) = f(t)k(s,z)$, in such a way that (4.2) embeds in a vector-valued polynomial Volterra equation, thus extending the Parker-Sochacki method to this setting and thereby obtaining a computationally efficient method for closely approximating solutions of (4.2).

Existence and uniqueness theorems for vector-valued Volterra integral equations are not common in the literature, so we begin Section 4.1 by stating and proving such a theorem. The proof is based on an application of the Contraction Mapping Theorem, so that we immediately obtain a theorem that guarantees convergence of Picard iterates to the unique solution. We continue Section 4.1 by describing how to introduce auxiliary variables to obtain a feasible, efficient computational procedure for approximating solutions of (4.2) when K factors as $K(t,s,z) = f(t)k(s,z)$. In Section 4.2 we illustrate the method with both linear and nonlinear examples.

4.1   Uniqueness Theorem

For a subset S of $\mathbb{R}^m$ we let $C(S,\mathbb{R}^n)$ denoted the set of continuous mappings from S into $\mathbb{R}^n$. The following theorem is a straightforward generalization to the vector-valued case of Theorem 2.1.1 of [?].

Theorem 4.1.1. *Let $I = [a,b] \subset \mathbb{R}$ and $J = \{(x,y) : x \in I, \ y \in [a,x]\} \subset I \times I$. Suppose $\varphi \in C(I,\mathbb{R}^n)$ and $K \in C(J \times \mathbb{R}^n, \mathbb{R}^n)$ and that K is Lipschitz in the last variable: there exists $L \in \mathbb{R}$ such that*

$$|K(x,y,\mathbf{z}) - K(x,y,\mathbf{z}')|_{sum} \leqslant L|\mathbf{z} - \mathbf{z}'|_{sum}$$

for all $(x,y) \in J$ and all $\mathbf{z},\mathbf{z}' \in \mathbb{R}^n$. Then the integral equation

$$\mathbf{y}(t) = \varphi(t) + \int_a^t K(t,s,\mathbf{y}(s))\,ds \tag{4.3}$$

has a unique solution $\mathbf{y}(t) \in C(I,\mathbb{R}^n)$.

Proof. Let C denote the set of continuous mappings from I into $\mathbb{R}^n$, $C = C(I,\mathbb{R}^n)$. Since φ and K are continuous, for any $\mathbf{y} \in C$ the right hand side of (4.3) is a continuous function of t. Thus we may define a mapping $\mathscr{T}$ from C into itself by

$$\mathscr{T}(\mathbf{y})(t) = \varphi(t) + \int_a^t K(t,s,\mathbf{y}(s))\,ds.$$

An element $\mathbf{y}$ of C is a solution of (4.3) if and only if $\mathbf{y}$ is a fixed point of the operator $\mathscr{T}$, which we will find by means of the Contraction Mapping Theorem 1.3.5.

Let $\boldsymbol{\eta}$ and $\boldsymbol{\mu}$ be elements of C. Then for all $t \in I$,

$$\begin{aligned}
|(\mathscr{T}\boldsymbol{\eta})(t) - (\mathscr{T}\boldsymbol{\mu})(t)|_{sum} &\leqslant \int_a^t |K(t,s,\boldsymbol{\eta}(s)) - K(t,s,\boldsymbol{\mu}(s))|_{sum}\,ds \\
&\leqslant \int_a^t L|\boldsymbol{\eta}(s) - \boldsymbol{\mu}(s)|_{sum}\,ds.
\end{aligned} \tag{4.4}$$

If we place, as usual, the supremum norm on C then we obtain from (4.4) the inequality

$$|(\mathscr{T}\boldsymbol{\eta})(t) - (\mathscr{T}\boldsymbol{\mu})(t)|_{sum} \leqslant L\|\boldsymbol{\eta} - \boldsymbol{\mu}\|_{sup}(b-a) \quad \text{for all } t \in I.$$

Taking the supremum over $t \in I$ shows that $\mathscr{T}$ is a contraction provided $L(b-a) < 1$, and since C is complete with respect to the supremum norm the theorem follows by the Contraction Mapping Theorem 1.3.5 , but with the additional condition that $L(b-a) < 1$. To obtain the theorem as stated (that is, without the extra condition that $L(b-a) < 1$) we fix any constant $\beta > 1$ and, following Hackbusch ([?]), instead equip C with the special

norm

$$||\mathbf{y}||_H = \max\{e^{-\beta L t}|y(t)|_{\text{sum}} : t \in I\}.$$

It is readily verified that $||\cdot||_H$ *does* define a norm on $\mathcal{C}$ and that it is equivalent to the supremum norm, hence $\mathcal{C}$ is also complete when equipped with the norm $||\cdot||_H$. Since, for any $\boldsymbol{\eta}$ and $\boldsymbol{\mu}$ in $\mathcal{C}$ and any $s \in I$,

$$
\begin{aligned}
|\boldsymbol{\eta}(s) - \boldsymbol{\mu}(s)|_{\text{sum}} &= e^{\beta L s} e^{-\beta L s} |\boldsymbol{\eta}(s) - \boldsymbol{\mu}(s)|_{\text{sum}} \\
&\leqslant e^{\beta L s} \max\{e^{-\beta L t}|\boldsymbol{\eta}(t) - \boldsymbol{\mu}(t)|_{\text{sum}} : t \in I\} \\
&= e^{\beta L s} ||\boldsymbol{\eta} - \boldsymbol{\mu}||_H,
\end{aligned}
$$

estimate (4.4) becomes

$$
\begin{aligned}
|(\mathscr{T}\boldsymbol{\eta})(t) - (\mathscr{T}\boldsymbol{\mu})(t)|_{\text{sum}} &\leqslant L ||\boldsymbol{\eta} - \boldsymbol{\mu}||_H \int_a^t e^{\beta L s}\, ds \\
&= L ||\boldsymbol{\eta} - \boldsymbol{\mu}||_H \tfrac{1}{\beta L}[e^{\beta L t} - e^{\beta L a}] \\
&= \tfrac{1}{\beta} e^{\beta L t}[1 - e^{\beta L(a-t)}] ||\boldsymbol{\eta} - \boldsymbol{\mu}||_H \\
&\leqslant \tfrac{1}{\beta} e^{\beta L t}[1 - e^{\beta L(a-b)}] ||\boldsymbol{\eta} - \boldsymbol{\mu}||_H
\end{aligned}
$$

for all $t \in I$. Multiplying by $e^{-\beta L t}$ and taking the supremum over $t \in I$ yields

$$||(\mathscr{T}\boldsymbol{\eta}) - (\mathscr{T}\boldsymbol{\mu})||_H \leqslant \left[\tfrac{1}{\beta}(1 - e^{-(b-a)\beta L})\right] ||\boldsymbol{\eta} - \boldsymbol{\mu}||_H.$$

By our choice $\beta > 1$ the constant in brackets is strictly less than 1 so that $\mathscr{T}$ is a contraction on $\mathcal{C}$ with respect to this norm, and the theorem follows by the Contraction Mapping Theorem 1.3.5. $\square$

Because the theorem was proved by means of the Contraction Mapping Theorem we immediately obtain the following result.

Theorem 4.1.2. *Under the hypotheses of Theorem* (4.1.1)*, for any choice of the initial mapping $y^{[0]}(t)$ the sequence of Picard iterates*

$$y^{[k+1]}(t) = \varphi(t) + \int_a^t K(t,s,y^{[k]}(s))\, ds$$

converges to the unique solution of the integral equation (4.3).

Now let a Volterra integral equation

$$y(t) = \varphi(t) + \int_a^t f(t)k(s,y(s))\, ds \tag{4.5}$$

be given, where $\varphi, f \in C([a,b], \mathbb{R})$, $k \in C([a,b] \times \mathbb{R}, \mathbb{R})$, and k satisfies a Lipschitz condition in y. Find functions $h_1(t), \ldots, h_r(t)$ and polynomials P and Q in r indeterminates, R in $r+2$ indeterminates, and $P_1, \ldots, P_r$ in $r+1$ indeterminates such that if $y(t)$ solves (4.5) then

(i) $\varphi(t) = P(h_1(t), \ldots, h_r(t))$,

(ii) $f(t) = Q(h_1(t), \ldots, h_r(t))$,

(iii) $k(s, y(s)) = R(s, y(s), h_1(s), \ldots, h_r(s))$, and

(iv) $v_j(t) = h_j(t)$ solves $v_j' = P_j(t, v_1, \ldots, v_r)$, $v_j(a) = h_j(a)$, $1 \leqslant j \leqslant r$.

The initial value problem obtained by adjoining to the system in (iv) the values of h_1 through h_r at a has a unique solution, which is the unique solution of the vector-valued Volterra equation

$$v_1 = v_1(a) + \int_a^t P_1(s, v_1(s), \ldots, v_r(s)) \, ds$$

$$\vdots \tag{4.6}$$

$$v_r = v_r(a) + \int_a^t P_r(s, v_1(s), \ldots, v_r(s)) \, ds.$$

Adjoin to (4.6) the original Volterra equation in the form

$$y(t) = P(v_1(t), \ldots, v_r(t)) + \int_a^t Q(v_1(t), \ldots, v_r(t)) R(s, y(s), v_1(s), \ldots, v_r(s)) \, ds$$

to obtain

$$y = P(v_1(t), \ldots, v_r(t)) + \int_a^t Q(v_1(t), \ldots, v_r(t)) R(s, y(s), v_1(s), \ldots, v_r(s)) \, ds$$

$$v_1 = v_1(a) + \int_a^t P_1(s, v_1(s), \ldots, v_r(s)) \, ds$$

$$\vdots \tag{4.7}$$

$$v_r = v_r(a) + \int_a^t P_r(s, v_1(s), \ldots, v_r(s)) \, ds.$$

System (4.7) satisfies the hypotheses of Theorem (4.1.1), hence has a unique solution, as does the original Volterra integral equation. Since $v_1, \ldots, v_r$ are completely specified by (4.1) and (4.6), the y component of the solution of the augmented Volterra integral equation (4.7) must be the solution of (4.5). But by Theorem 4.1.2 the Picard iteration scheme applied to (4.7), say with $y^{[0]}(t) \equiv y(0)$, converges and is computationally feasible, so we obtain a computable approximation to the solution of (4.5).

4.2 Examples

In this section we illustrate the method by means of several examples which include both linear and nonlinear Volterra integral equations. A Volterra equation (4.2) is said to be linear if the integrand factors as $K(t, s, y) = k(t, s)y$ and nonlinear if $K(t, s, y) = k(t, s)f(s, y)$, where $f(s, y)$ is a nonlinear function of y.

Example 4.2.1. Effati ([?]) introduced the linear Volterra integral equation of the second kind

$$y(t) = \exp t \sin t + \int_0^t \frac{2 + \cos t}{2 + \cos s} y(s) \, ds, \tag{4.8}$$

which is of the form (4.5) with $\varphi(t) = \exp t \sin t$, $f(t) = 2 + \cos t$, and the kernel of a form we can treat, $k(s, y(s)) = y(s)/(2 + \cos s)$.

One appropriate choice of auxiliary variables is

$$v_1 = \exp(t)$$

$$v_2 = \cos(t)$$

$$v_3 = \sin(t)$$

$$v_4 = 2 + v_2$$

$$v_5 = \frac{1}{v_4},$$

which satisfy the system of first order ordinary differential equations

$$v_1' = v_1$$

$$v_2' = -v_3$$

$$v_3' = v_2$$

$$v_4' = v_2' = -v_3$$

$$v_5' = \frac{-v_4'}{v_4^2} = v_3 v_5^2,$$

which is equivalent to

$$v_1(t) = v_1(0) + \int_0^t v_1(s) \, ds$$

$$v_2(t) = v_2(0) - \int_0^t v_3(s) \, ds$$

$$v_3(t) = v_3(0) + \int_0^t v_2(s) \, ds$$

$$v_4(t) = v_4(0) - \int_0^t v_3(s) \, ds$$

$$v_5(t) = v_5(0) + \int_0^t v_3(s) v_5^2 \, ds.$$

The initial values of the auxiliary variables are determined by their definition. The initial value $y(0)$ of the solution of the integral equation (4.8) is found simply by evaluating that equation at $t = 0$ to obtain $y(0) = 0$.

Thus the iteration scheme is

$$y^{[k+1]}(t) = v_1^{[k]} v_3^{[k]} + v_4^{[k]} \int_0^t v_5^{[k]} y^{[k]} ds$$

$$v_1^{[k+1]}(t) = 1 + \int_0^t v_1^{[k]} ds$$

$$v_2^{[k+1]}(t) = 1 - \int_0^t v_3^{[k]} ds$$

$$v_3^{[k+1]}(t) = \int_0^t v_2^{[k]} ds$$

$$v_4^{[k+1]}(t) = \int_0^t v_3^{[k]} ds$$

$$v_5^{[k+1]}(t) = \frac{1}{3} + \int_0^t v_3^{[k]} (v_5^{[k]})^2 ds$$

We can initialize as we please, but it is reasonable to choose

$$y^{[0]} \equiv 0$$

$$v_1^{[0]} \equiv \exp 0 = 1$$

$$v_2^{[0]} \equiv \cos 0 = 1$$

$$v_3^{[0]} \equiv \sin 0 = 0$$

$$v_4^{[0]} \equiv 2 + v_2^{[0]} = 3$$

$$v_5^{[0]} \equiv \frac{1}{3}.$$

The exact solution of (4.8) is

$$y(t) = \exp t \sin t + \exp t \left(2 + \cos t \right) \left(\ln 3 - \ln \left(2 + \cos t \right) \right),$$

whose Maclaurin series, with its coefficients rounded to five decimal places, begins

$$y(t) = 1.00000t + 1.50000t^2 + 0.83333t^3 + 0.16667t^4 - 0.03333t^5 - 0.02593t^6 - 0.00529t^7 + O(t^8).$$

The Maclaurin series of the eighth Picard iterate, $y^{[8]}(t)$, with its coefficients rounded to five decimal places, begins

$$y(t) = 1.00000t + 1.50000t^2 + 0.83333t^3 + 0.16667t^4 - 0.03333t^5 - 0.02593t^6 - 0.00529t^7 + O(t^8).$$

A plot of the error in the approximation of the exact solution by $y^{[8]}(t)$ is given in Figure 4.1.

Example 4.2.2. In [?] Biazar introduced the nonlinear Volterra integral equation of the second kind

$$y(t) = \tfrac{1}{2}\sin 2t + \int_0^t \tfrac{3}{2}y(s)^2\cos(s-t)\,ds. \tag{4.9}$$

To fit this into the framework of (4.5) we begin by applying the cosine difference identity $\cos(t-s) = \cos s\cos t + \sin s\sin t$, obtaining

$$y(t) = \tfrac{1}{2}\sin 2t + \tfrac{3}{2}\left(\cos t\int_0^t y(s)^2\cos s\,ds + \sin t\int_0^t y(s)^2\sin s\,ds\right).$$

Introducing the auxiliary variables $v = \cos s$ and $w = \sin s$, which solve the system

$$v' = -w, \quad w' = v,$$

upon integration we obtain the equivalent system of integral equations

$$v(t) = v(0) - \int_0^t w(s)\,ds$$
$$w(t) = w(0) + \int_0^t v(s)\,ds.$$

The initial values of the auxiliary variables are determined by their definition. The initial value $y(0)$ of the solution of the integral equation (4.9) is found simply by evaluating that equation at $t = 0$ to obtain $y(0) = 0$. Thus the iteration scheme is

$$y^{[k+1]}(t) = w^{[k]}v^{[k]} + \tfrac{3}{2}\left(v^{[k]}(t)\int_0^t v^{[k]}(s)(y^{[k]})^2(s)\,ds + w^{[k]}(t)\int_0^t w^{[k]}(s)(y^{[k]})^2(s)\,ds\right)$$
$$w^{[k+1]}(t) = 0 + \int_0^t v^{[k]}(s)\,ds$$
$$v^{[k+1]}(t) = 1 - \int_0^t w^{[k]}(s)\,ds.$$

We initialize with

$$y^{[0]}(t) \equiv y(0) = 0$$
$$w^{[0]}(t) \equiv \sin 0 = 0$$
$$v^{[1]}[0](t) \equiv \cos 0 = 1.$$

The exact solution of (4.9) is $y(t) = \sin t$, whose Maclaurin series, with its coefficients rounded to five decimal places, begins

$$y(t) = 1.00000t - 0.16667t^3 + 0.00833t^5 - 0.00020t^7 + O(t^9).$$

The Maclaurin series of the eighth Picard iterate, $y^{[8]}(t)$, with its coefficients rounded to five decimal places, begins

$$y^{[8]}(t) = 1.00000t - 0.16667t^3 + 0.008333t^5 + 0.00000t^7 + O(t^9).$$

A plot of the error in the approximation of the exact solution by $y^{[8]}(t)$ is given in Figure 4.2.

Example 4.2.3. As a final and somewhat more elaborate example consider the linear Volterra integral equation of the second kind given by

$$y(t) = \tan t - \tfrac{1}{4}\sin 2t - \tfrac{1}{2}t + \int_0^t \frac{1}{1+y^2(s)}\,ds. \tag{4.10}$$

This is a corrected version of an integral equation given by Vahidian and his colleagues in [?]. As explained in the first paragraph of this chapter, because the integral part is independent of t, (4.10) must be equivalent to an initial value problem. In fact the equivalent problem is

$$y'(t) = \sec^2 t - \tfrac{1}{2}\cos 2t - \tfrac{1}{2} + \frac{1}{1+y^2(t)}, \quad y(0) = 0,$$

which was obtained by differentiating (4.10) to get the differential equation and by evaluating (4.10) at $t = 0$ to get the initial value. Of course by means of the identity $\cos^2 t = \tfrac{1}{2}(1 + \cos 2t)$ the differential equation can be more compactly expressed as

$$y'(t) = \sec^2 t - \cos^2 t + \frac{1}{1+y^2(t)}, \tag{4.11}$$

which will be important later.

To approximate the unique solution of (4.10) we introduce the auxiliary variables

$$v_1(t) = \sin t$$
$$v_2(t) = \cos t$$
$$v_3(t) = \frac{1}{v_2}$$
$$v_4(t) = 1 + y^2,$$
$$v_5(t) = \frac{1}{v_4}.$$

Note that in contrast with the previous examples the unknown function $y(t)$ figures into the definition of some of these variables, but in a polynomial way. Thus when we compute their derivatives y also appears. Thanks to (4.11), it does so in a polynomial way, since by that identity $y' = v_3^2 - v_2^2 + v_5$ and we have additionally

$$v_1' = v_2$$
$$v_2' = -v_1$$
$$v_3' = v_1 v_3^2$$
$$v_4' = 2y(v_3^2 - v_2^2 + v_5)$$
$$v_5' = -2y v_5^2 (v_3^2 - v_2^2 + v_5).$$

This system of ordinary differential equations, together with the equation satisfied by y' and the known initial values of all the variables involved, is equivalent to the system of integral equations

$$y(t) = v_1 v_3 - \tfrac{1}{2} v_1 v_2 - \tfrac{1}{2} t + \int_0^t v_5(s)\, ds$$
$$v_1(t) = \int_0^t v_2(s)\, ds$$
$$v_2(t) = 1 - \int_0^t v_1(s)\, ds$$
$$v_3(t) = 1 + \int_0^t v_1(s) v_3^2(s)\, ds$$
$$v_4(t) = 1 + 2 \int_0^t y(s)(v_3^2(s) - v_2^2(s) + v_5(s))\, ds$$
$$v_5(t) = 1 - 2 \int_0^t y(s) v_5^2(s)(v_3^2(s) - v_2^2(s) + v_5(s))\, ds.$$

Setting up the obvious iteration scheme based on these integral equations, and initializing with the constant functions $y(t) \equiv y(0)$ and $v_j(t) \equiv v_j(0)$ for $j = 1, 2, 3, 4, 5$, the Picard iterate $y^{[28]}(t)$ with coefficients rounded to five decimal places is

$$y(t) = 1.00000t + 0.33333t^3 + 0.13333t^5 + 0.05397t^7 + 0.02187t^9 + 0.00886t^{11} + O(t^{13}).$$

The exact solution is $y(t) = \tan t$, whose Maclaurin series, with coefficients rounded to five decimal places, is

$$y(t) = 1.00000t + 0.33333t^3 + 0.13333t^5 + 0.05397t^7 + 0.02187t^9 + 0.00886t^{11} + O(t^{13}).$$

A plot of the error in the approximation of the exact solution by $y^{[28]}(t)$ is given in Figure 4.3.
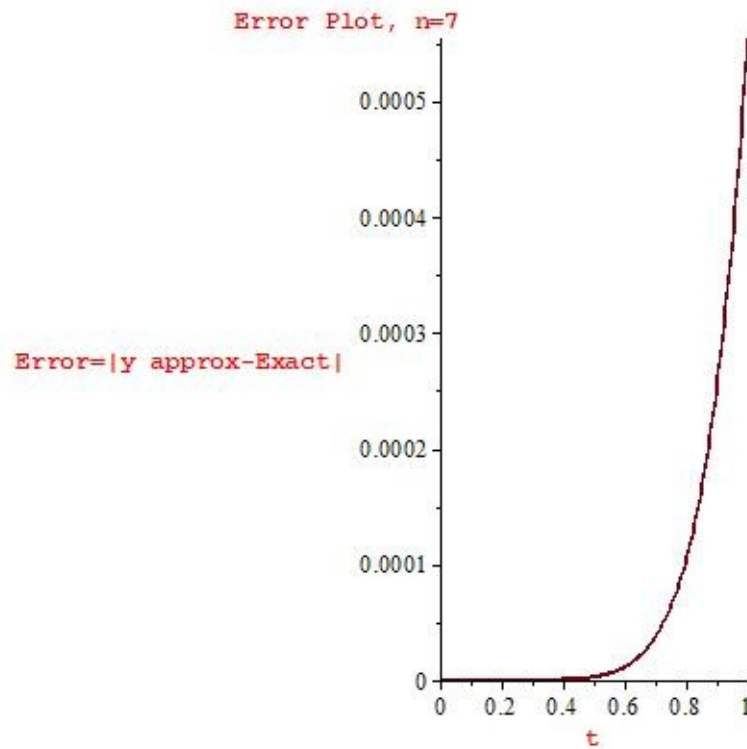
Figure 4.1: The error plot for Volterra Integral, $|y(t) - y^{[8]}|$ with $n = 7$ iterations, where $y(t)$ is the exact solution and $y^{[8]}(t)$ is the approximate solution.

Figure 4.2: Error plot for nonlinear Volterra Integral, $|y(t) - y^{[8]}(t)|$ with $n = 8$ iterations, where $y(t)$ is the exact solution and $y^{[8]}(t)$ is the approximate solution.

.

The error plot for linear Volterra Integral, $|y(t) - y[28]|$

Figure 4.3: The error plot for linear Volterra Integral, $|y(t) - y^{[28]}(t)|$ with $n = 27$ iterations, where $y(t)$ is the exact solution and $y^{[28]}(t)$ is the approximate solution.
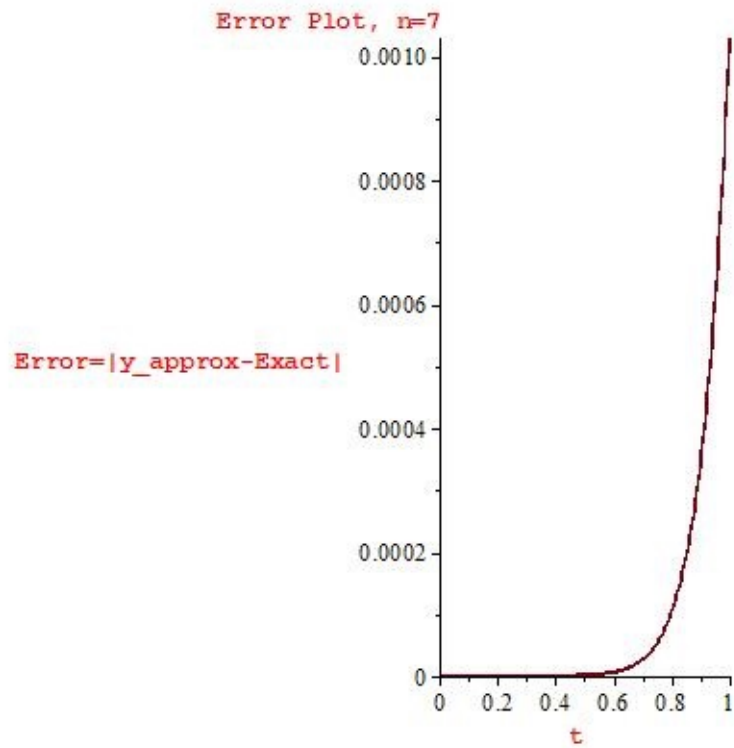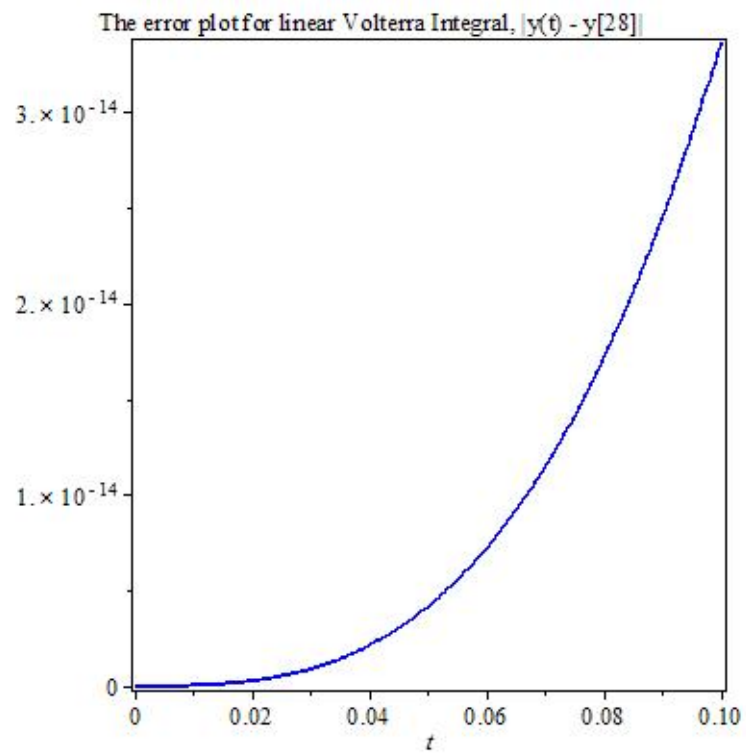
CHAPTER 5: A MORE GENERAL ALGORITHM FOR APPROXIMATING SOLUTIONS OF
BOUNDARY VALUE PROBLEMS

In Chapter 2 we presented an efficient algorithm for approximating solutions of two-point boundary value problems of the form

$$y'' = f(t, y, y'), \qquad y(a) = \alpha, \quad y(b) = \beta. \tag{5.1}$$

In this chapter we develop a modification of the algorithm that can extend its scope. The idea is to partition the interval $[a, b]$ into n subintervals and simultaneously and recursively approximate the solutions to the n boundary value problems that are induced on the subintervals by (5.1) and its solution. We begin in Section 5.1 by illustrating the ideas involved in the simplest case, $n = 2$, where the key ideas in the modified method are not obscured by the details and are thus most easily understood. In this first section we will also introduce notation and several lemmas that will be useful later. In Section 5.2 we present the modified algorithm in the general case and state and prove a theorem that guarantees its success. Finally, in Section 5.3 we illustrate the method with several examples.

Throughout this chapter we will assume that the function f is continuous and Lipschitz in the last two variables with Lipschitz constant L.

5.1   The Simplest Case

We begin by recalling from Chapter 1 the following fact, which for ease of future reference we separate out as a lemma, but without a detailed statement of hypotheses (see (1.1) and (1.2) and Lemma 1.3.2).

Lemma 5.1.1. *The one-dimensional two-point boundary value problem*

$$y'' = f(t, y, y'), \qquad y(c) = \lambda, \quad y(d) = \delta$$

is equivalent to the vector integral equation

$$\begin{pmatrix} y(t) \\ u(t) \end{pmatrix} = \begin{pmatrix} \lambda \\ \gamma \end{pmatrix} + \int_c^t \begin{pmatrix} u(s) \\ f(s, y(s), u(s)) \end{pmatrix} ds \tag{5.2a}$$

$$\gamma = \frac{1}{d-c} \left[\delta - \lambda - \int_c^d (d-s) f(s, y(s), u(s)) \, ds \right]. \tag{5.2b}$$

It will be convenient to solve (5.2b) for λ, yielding

$$\lambda = \delta - (d-c)\gamma - \int_c^d (d-s)f(s,y(s),u(s))\,ds. \tag{5.3}$$

We also note that if $\gamma_1 = y'(c)$ and $\gamma_2 = y'(d)$ then by the Fundamental Theorem of Calculus

$$\gamma_2 = \gamma_1 + \int_c^d f(s,y(s),u(s))\,ds. \tag{5.4}$$

For the following discussion we speak as if it were certain that problem (5.1) has a unique solution, which we denote $y(t) = \eta(t)$.

Partition the interval $[a,b]$ into two subintervals of equal length $h = (b-a)/2$ and let the partition points be denoted $a = t_0 < t_1 < t_2 = b$. Then (5.1) and its solution $\eta(t)$ induce two boundary value problems,

$$y'' = f(t,y,y'), \qquad y(t_0) = \alpha, \quad y(t_1) = \eta(t_1) \tag{5.5}$$

on $[t_0, t_1]$ and

$$y''(t) = f(t,y,y'), \qquad y(t_1) = \eta(t_1), \quad y(t_2) = \beta \tag{5.6}$$

on $[t_1, t_2]$. Letting $\beta_0 = \alpha$, $\beta_1 = \eta(t_1)$, $\gamma_1 = \eta'(t_0)$, and $\gamma_2 = \eta'(t_1)$, the solutions of these boundary value problems are the solutions of the initial value problems

$$y'' = f(t,y(t),y'(t)), \qquad y(t_0) = \beta_0, \quad y'(t_0) = \gamma_1 \tag{5.7}$$

and

$$y'' = f(t,y(t),y'(t)), \qquad y(t_1) = \beta_1, \quad y'(t_1) = \gamma_2, \tag{5.8}$$

which we express as the systems of first order equations

$$\begin{cases} y' = u & y(t_0) = \beta_0 \\ u' = f(t,y,u) & u(t_0) = \gamma_1 \end{cases} \qquad \text{or} \qquad \mathbf{y} = \mathbf{f}(t,\mathbf{y}), \ \mathbf{y}(t_0) = \begin{pmatrix} \beta_0 \\ \gamma_1 \end{pmatrix} \tag{5.7'}$$
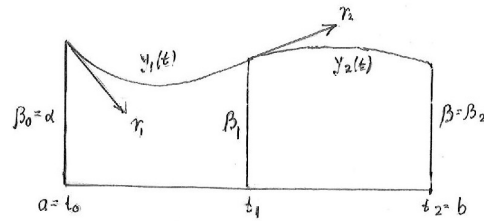
and

$$\begin{cases} y' = u & y(t_1) = \beta_1 \\ u' = f(t,y,u) & u(t_1) = \gamma_2 \end{cases} \qquad \text{or} \qquad \mathbf{y} = \mathbf{f}(t,\mathbf{y}), \ \mathbf{y}(t_1) = \begin{pmatrix} \beta_1 \\ \gamma_2 \end{pmatrix} \tag{5.8'}$$

Let

$$\mathbf{y}_1(t) = \begin{pmatrix} y_1(t) \\ u_1(t) \end{pmatrix} \qquad \text{and} \qquad \mathbf{y}_2(t) = \begin{pmatrix} y_2(t) \\ u_2(t) \end{pmatrix}$$

denote the solutions of the initial value problems (5.7′) and (5.8′), respectively. Then $\eta(t)$ is the concatenation of $y_1(t)$ and $y_2(t)$. See Figure (5.1)..

The idea now is to apply the algorithm of Chapter 2 to each of the boundary value problems (5.5) and (5.6) simultaneously, coupling the work on the separate intervals at each step. That is, we wish to recursively approximate the vector functions $\mathbf{y}_1(t)$ and $\mathbf{y}_2(t)$ and the constants γ_1, γ_2, and β_1 better and better, updating each one on each pass through the recursion.

Choose any initialization of the unknown functions and constants,

$$y_1^{[0]}(t), \quad u_1^{[0]}(t), \quad y_2^{[0]}(t), \quad u_2^{[0]}(t), \quad \gamma_1^{[0]}, \quad \gamma_2^{[0]}, \quad \beta_1^{[0]}.$$

We can update the approximations of $y_1^{[0]}(t)$, $u_1^{[0]}(t)$, $y_2^{[0]}(t)$, and $u_2^{[0]}(t)$ using the right hand side of (5.2a) applied to each subinterval. Writing

$$f_j(s) \overset{\text{def}}{=} f(s, y_j(s), u_j(s)), \quad j = 1, 2,$$

we update the approximation of γ_2 in terms of the approximation of γ_1 using (5.4) applied on $[t_1, t_2]$, namely,

$$\gamma_2 = \gamma_1 + \int_{t_0}^{t_1} f_1(s) \, ds. \tag{5.9}$$

We update the approximation of β_1 in terms of β and the approximation of γ_2 using (5.3), namely

$$\beta_1 = \beta - h\gamma_2 - \int_{t_1}^{t_2} (t_2 - s) f_2(s) \, ds. \tag{5.10}$$

Finally, we update the approximation of γ_1 in terms of α and the approximation of β_1 using (5.2b), namely

$$\gamma_1 = \frac{1}{h} \left[\beta_1 - \alpha - \int_{t_0}^{t_1} (t_1 - s) f_1(s) \, ds \right]. \tag{5.11}$$

Introducing the simplifying notation

$$f_j^{[r]}(s) \overset{\text{def}}{=} f(s, y_j^{[r]}(s), u_j^{[r]}(s)), \quad j = 1, 2, \quad r \in \mathbb{Z}^+ \cup \{0\},$$

and the natural shorthand $\mathbf{y}_j^{[r]}(t)$ the recurrence formulas are

$$\mathbf{y}_1^{[k+1]}(t) \overset{\text{def}}{=} \begin{pmatrix} y_1^{[k+1]}(t) \\ u_1^{[k+1]}(t) \end{pmatrix} = \begin{pmatrix} \alpha \\ \gamma_1^{[k]} \end{pmatrix} + \int_{t_0}^{t} \begin{pmatrix} u_1^{[k]} \\ f_1^{[k]}(s) \end{pmatrix} ds \tag{5.12a}$$

$$\mathbf{y}_2^{[k+1]}(t) \overset{\text{def}}{=} \begin{pmatrix} y_2^{[k+1]}(t) \\ u_2^{[k+1]}(t) \end{pmatrix} = \begin{pmatrix} \beta_1^{[k]} \\ \gamma_2^{[k]} \end{pmatrix} + \int_{t_1}^{t} \begin{pmatrix} u_2^{[k]} \\ f_2^{[k]}(s) \end{pmatrix} ds \tag{5.12b}$$

$$\gamma_2^{[k+1]} = \gamma_1^{[k]} + \int_{t_0}^{t_1} f_1^{[k+1]}(s) \, ds \tag{5.12c}$$

$$\beta_1^{[k+1]} = \beta - h\gamma_2^{[k+1]} - \int_{t_1}^{t_2} (t_2 - s) f_2^{[k+1]}(s) \, ds \tag{5.12d}$$

$$\gamma_1^{[k+1]} = \frac{1}{h} \left[\beta_1^{[k+1]} - \alpha - \int_{t_0}^{t_1} (t_1 - s) f_1^{[k+1]}(s) \, ds \right]. \tag{5.12e}$$

Clearly if $\mathbf{y}_j^{[k]}(t)$, $j = 1, 2$, converge, then $\gamma_2^{[k]}$, $\beta_1^{[k]}$, and $\gamma_1^{[k]}$ must converge. If the convergence is uniform then we may pass the limits into the integrals to obtain the fact that the limit functions $y_1(t)$ and $y_2(t)$ solve the

ordinary differential equations in (5.7) and (5.8) and that

$$\Gamma_j \stackrel{\text{def}}{=} \lim_{k \to \infty} \gamma_j^{[k]}, j = 1, 2 \quad \text{and} \quad B_1 \stackrel{\text{def}}{=} \lim_{k \to \infty} \beta_1^{[k]}$$

satisfy

$$\Gamma_2 = \Gamma_1 + \int_{t_0}^{t_1} f_1(s) \, ds \tag{5.13}$$

$$B_1 = \beta - h\Gamma_2 - \int_{t_1}^{t_2} (t_2 - s) f_2(s) \, ds \tag{5.14}$$

$$\Gamma_1 = \frac{1}{h} \left[B_1 - \alpha - \int_{t_0}^{t_1} (t_1 - s) f_1(s) \, ds \right]. \tag{5.15}$$

It is intuitively clear that the concatenation $y(t)$ of the limits $y_1(t)$ and $y_2(t)$ solves the original boundary value problem (5.1), but here are the details. Evaluating the limit of (5.12a) at $t = t_0$ gives

$$y_1(t_0) = \alpha \tag{5.16}$$

and

$$y_1'(t_0) = \Gamma_1. \tag{5.17}$$

Inserting (5.17) into (5.15) gives

$$y_1'(t_0) = \frac{1}{h} \left[B_1 - \alpha - \int_{t_0}^{t_1} (t_1 - s) f(s, y_1(s), u_1(s)) \, ds \right]$$

so that by (5.2b)

$$B_1 = y_1(t_1). \tag{5.18}$$

Evaluating the limit of (5.12b) at $t = t_1$ gives

$$B_1 = y_2(t_1) \tag{5.19}$$

and

$$\Gamma_2 = y_2'(t_1). \tag{5.20}$$

Equations (5.18) and (5.19) give

$$y_1(t_1) = y_2(t_1). \tag{5.21}$$

Inserting (5.19) and (5.20) into (5.14) gives

$$y_2(t_1) = \beta - hy_2'(t_1) - \int_{t_1}^{t_2} (t_2 - s)f(s, y_2(s), u_2(s)) \, ds$$

so that by (5.3)

$$y_2(t_2) = \beta. \tag{5.22}$$

Inserting (5.20) and (5.17) into (5.13) gives

$$y_2'(t_1) = y_1'(t_0) + \int_{t_0}^{t_1} f(s, y_1(s), u_1(s)) \, ds$$

which by the Fundamental Theorem gives

$$y_2'(t_1) = y_1'(t_1). \tag{5.23}$$

Equations (5.16), (5.21), (5.22), and (5.23) imply that the concatenation $y(t)$ of $y_1(t)$ and $y_2(t)$ is a C^1 function on $[a,b]$ that has a second derivative on $[a,t_1) \cup (t_1,b]$ that satisfies $y''(t) = f(t, y(t), y'(t))$ on $[a,t_1) \cup (t_1,b]$. Since the right hand side is continuous on $[a,b]$ by Lemma 5.1.4, whose statement and proof are deferred to the end of this section, $y''(t)$ exist on $[a,b]$, hence $y(t)$ solves the original boundary value problem (5.1).

In order to prove convergence of the sequence of iterates $\mathbf{y}_j^{[k]}(t)$, $j = 1,2$, we need estimates on $|\gamma_j^{[k+1]} - \gamma_j^{[k]}|$, $j = 1,2$ and on $|\beta_1^{[k+1]} - \beta_1^{[k]}|$. It is clear that we can estimate $|\gamma_2^{[k+1]} - \gamma_2^{[k]}|$ in terms of $|\gamma_1^{[k+1]} - \gamma_1^{[k]}|$ and $|\gamma_1^{[k+1]} - \gamma_1^{[k]}|$ in terms of $|\beta_1^{[k+1]} - \beta_1^{[k]}|$, but the estimate on $|\beta_1^{[k+1]} - \beta_1^{[k]}|$ involves $|\gamma_2^{[k+1]} - \gamma_2^{[k]}|$, hence circularity. An idea for a way out is to insert the expression for γ_1 in the last recurrence equation into that for γ_2, insert the resulting expression for γ_2 into the expression for β_1, and solve the resulting equation for β_1. However, the step-indices on β_1 do not match and the procedure yields an equation with $\beta_1^{[k+1]}$ on the left and $\beta_1^{[k]}$ on the right. A solution to this dilemma is to follow the same procedure, not with the recurrence equations, but with the equations (5.9) through (5.11) on which they are based. That is, insert the expression for γ_1 in (5.11) into (5.9), insert the resulting expression for γ_2 into (5.10), and solve the resulting equation for β_1. The result is

$$\beta_1 = \tfrac{1}{2}\left[\beta + \alpha + \int_{t_0}^{t_1}(t_1 - s)f_1(s)\,ds - h\int_{t_0}^{t_1} f_1(s)\,ds - \int_{t_1}^{t_2}(t_2 - s)f_2(s)\,ds\right]. \tag{5.24}$$

We base the update of the approximation of β_1 on this equation. Thus the recursion for which we will prove

convergence is (5.12) but with (5.12d) replaced by

$$\beta_1^{[k+1]} = \tfrac{1}{2}\left[\beta + \alpha + \int_{t_0}^{t_1}(t_1 - s)f_1^{[k+1]}(s)\,ds - h\int_{t_0}^{t_1}f_1^{[k+1]}(s)\,ds - \int_{t_1}^{t_2}(t_2 - s)f_2^{[k+1]}(s)\,ds\right]. \qquad (5.12\text{d}')$$

Recall that we need concern ourselves only with the convergence of the vector-valued functions $\mathbf{y}_1(t)$ and $\mathbf{y}_2(t)$ on $[t_0,t_1]$ and $[t_1,t_2]$ but that the convergence must be uniform in order to prove that we have found a solution of the original boundary value problem. Thus we are concerned with the sequence $(\mathbf{y}_1(t),\mathbf{y}_2(t))$ in $C([t_0,t_1],\mathbb{R}^2) \times C([t_1,t_2],\mathbb{R}^2)$. We place the sup norm, the norm of uniform convergence, on each of the function spaces, and do so with respect to the sum norm on $\mathbb{R}^2$ so that the norm can be brought inside the integral. On the product of the function spaces we place the maximum norm. Thus for $(\mathbf{u}(t),\mathbf{v}(t)) \in C([t_0,t_1],\mathbb{R}^2) \times C([t_1,t_2],\mathbb{R}^2)$

$$||(\mathbf{u}(t),\mathbf{v}(t))||_{\max} = \max\{||\mathbf{u}(t)||_{\sup},||\mathbf{v}(t)||_{\sup}\}$$

and

$$||\mathbf{u}(t)||_{\sup} = \left|\left|\begin{pmatrix}u_1(t)\\u_2(t)\end{pmatrix}\right|\right|_{\sup} = \sup\{|u_1(t)| + |u_2(t)| : t \in [t_0,t_1]\}.$$

These choices are illustrated by the routine computations in the proof of Lemma 5.1.3, which gives two estimates that will be used repeatedly and which appears at the end of this section, where it has been collected with other such results.

As a preliminary to proving the convergence of $\mathbf{y}_1^{[k]}(t)$ and $\mathbf{y}_2^{[k]}(t)$ we examine the sequences of constants $\beta_1^{[k]}$, $\gamma_1^{[k]}$, and $\gamma_2^{[k]}$. First using (5.12d$'$) and recalling that f is Lipschitz in its last two entries with Lipschitz constant L we estimate, appealing to Lemma 5.1.3 for the last inequality,

$$|\beta_1^{[k+1]} - \beta_1^{[k]}|$$
$$\leqslant \tfrac{1}{2}\left[\int_{t_0}^{t_1}(t_1 - s)|f_1^{[k+1]}(s) - f_1^{[k]}(s)|\,ds + h\int_{t_0}^{t_1}|f_1^{[k+1]}(s) - f_1^{[k]}(s)|\,ds\right.$$
$$\left. + \int_{t_1}^{t_2}(t_2 - s)|f_2^{[k+1]}(s) - f_2^{[k]}(s)|\,ds\right]$$
$$\leqslant \tfrac{1}{2}\left[L\int_{t_0}^{t_1}(t_1 - s)|\mathbf{y}_1^{[k+1]}(s) - \mathbf{y}_1^{[k]}(s)|\,ds + hL\int_{t_0}^{t_1}|\mathbf{y}_1^{[k+1]}(s) - \mathbf{y}_1^{[k]}(s)|\,ds\right.$$
$$\left. + L\int_{t_1}^{t_2}(t_2 - s)|\mathbf{y}_2^{[k+1]}(s) - \mathbf{y}_2^{[k]}(s)|\,ds\right]$$
$$\leqslant \tfrac{1}{2}L\left[\tfrac{1}{2}h^2 + h^2 + \tfrac{1}{2}h^2\right]||\mathbf{y}^{[k+1]} - \mathbf{y}^{[k]}||_{\max}$$
$$= Lh^2||\mathbf{y}^{[k+1]} - \mathbf{y}^{[k]}||_{\max}.$$

Using (5.12e), this estimate, and Lemma 5.1.3 we obtain

$$\begin{aligned}
|\gamma_1^{[k+1]} - \gamma_1^{[k]}| &\leqslant \tfrac{1}{h}|\beta_1^{[k+1]} - \beta_1^{[k]}| + \tfrac{1}{h}\int_{t_0}^{t_1}(t_1 - s)|f_1^{[k+1]}(s) - f_1^{[k]}(s)|\,ds \\
&\leqslant Lh||\mathbf{y}^{[k+1]} - \mathbf{y}^{[k]}||_{\max} + \tfrac{1}{2}hL||\mathbf{y}^{[k+1]} - \mathbf{y}^{[k]}||_{\max} \\
&= \tfrac{3}{2}Lh||\mathbf{y}^{[k+1]} - \mathbf{y}^{[k]}||_{\max}
\end{aligned}$$

which in turn with (5.12c) and Lemma 5.1.3 yields

$$\begin{aligned}
|\gamma_2^{[k+1]} - \gamma_2^{[k]}| &\leqslant |\gamma_1^{[k+1]} - \gamma_1^{[k]}| + |\int_{t_0}^{t_1}|f_1^{[k+1]}(s) - f_1^{[k]}(s)|\,ds \\
&\leqslant \tfrac{3}{2}Lh||\mathbf{y}^{[k+1]} - \mathbf{y}^{[k]}||_{\max} + Lh||\mathbf{y}^{[k+1]} - \mathbf{y}^{[k]}||_{\max} \\
&= \tfrac{5}{2}Lh||\mathbf{y}^{[k+1]} - \mathbf{y}^{[k]}||_{\max}.
\end{aligned}$$

Turning now to the sequences $\mathbf{y}_1^{[k]}(t)$ and $\mathbf{y}_2^{[k]}(t)$, using these estimates and Lemma 2.1.2, for any $t \in [t_0, t_1]$

$$\begin{aligned}
|\mathbf{y}_1^{[k+1]}(t) - \mathbf{y}_1^{[k]}(t)| &\leqslant \left|\begin{pmatrix}\alpha\\\gamma_1^{[k]}\end{pmatrix} - \begin{pmatrix}\alpha\\\gamma_1^{[k-1]}\end{pmatrix}\right|_{\text{sum}} + \int_{t_0}^{t}\left|\begin{pmatrix}u_1^{[k]}(s)\\f_1^{[k]}(s)\end{pmatrix} - \begin{pmatrix}u_1^{[k-1]}(s)\\f_1^{[k-1]}(s)\end{pmatrix}\right|_{\text{sum}}\,ds \\
&= |\gamma_1^{[k]} - \gamma_1^{[k-1]}| + (1+L)h||\mathbf{y}_1^{[k]} - \mathbf{y}_1^{[k-1]}||_{\sup} \\
&\leqslant \tfrac{3}{2}Lh||\mathbf{y}^{[k+1]} - \mathbf{y}^{[k]}||_{\max} + (1+L)h||\mathbf{y}^{[k]} - \mathbf{y}^{[k-1]}||_{\max} \\
&= (1 + \tfrac{5}{2}L)h||\mathbf{y}^{[k]} - \mathbf{y}^{[k-1]}||_{\max}
\end{aligned}$$

hence

$$||\mathbf{y}_1^{[k+1]} - \mathbf{y}_1^{[k]}||_{\sup} \leqslant (1 + \tfrac{5}{2}L)h||\mathbf{y}^{[k]} - \mathbf{y}^{[k-1]}||_{\max}$$

and, assuming that $h^2 \leqslant h$, for any $t \in [t_1, t_2]$

$$\begin{aligned}
|\mathbf{y}_2^{[k+1]}(t) - \mathbf{y}_2^{[k]}(t)|_{\text{sum}} &\leqslant \left|\begin{pmatrix}\beta_1^{[k]}\\\gamma_2^{[k]}\end{pmatrix} - \begin{pmatrix}\beta_1^{[k-1]}\\\gamma_2^{[k-1]}\end{pmatrix}\right|_{\text{sum}} + \int_{t_1}^{t}\left|\begin{pmatrix}u_2^{[k]}(s)\\f_2^{[k]}(s)\end{pmatrix} - \begin{pmatrix}u_2^{[k-1]}(s)\\f_2^{[k-1]}(s)\end{pmatrix}\right|_{\text{sum}}\,ds \\
&= |\beta_1^{[k]} - \beta_1^{[k-1]}| + |\gamma_2^{[k]} - \gamma_2^{[k-1]}| + (1+L)h||\mathbf{y}_2^{[k]} - \mathbf{y}_2^{[k-1]}||_{\sup} \\
&\leqslant [Lh^2 + \tfrac{5}{2}Lh + (1+L)h]||\mathbf{y}^{[k]} - \mathbf{y}^{[k-1]}||_{\max} \\
&= (1 + \tfrac{9}{2}L)h||\mathbf{y}^{[k]} - \mathbf{y}^{[k-1]}||_{\max}
\end{aligned}$$

hence

$$||\mathbf{y}_2^{[k+1]} - \mathbf{y}_2^{[k]}||_{\sup} \leqslant (1 + \tfrac{9}{2}L)h\,||\mathbf{y}^{[k]} - \mathbf{y}^{[k-1]}||_{\max}.$$

Thus if $(1 + \tfrac{9}{2}L)h < 1$ then by Lemma 5.1.2 below the sequence

$$\mathbf{y}^{[k]}(t) = (\mathbf{y}_1^{[k]}(t), \mathbf{y}_2^{[k]}(t)) = \left(\begin{pmatrix} y_1^{[k]}(s) \\ u_1^{[k]}(s) \end{pmatrix}, \begin{pmatrix} y_2^{[k]}(s) \\ u_2^{[k]}(s) \end{pmatrix} \right)$$

is a Cauchy sequence in the complete space $C([t_0,t_1],\mathbb{R}^2) \times C([t_1,t_2],\mathbb{R}^2)$, hence converges, hence the component functions converge uniformly on $[t_0,t_1]$ and $[t_1,t_2]$. This condition was derived on the assumption that $h \leqslant 1$. If it fails we simply replace h by h^2 in the last two estimates to obtain the similar sufficient condition $(1 + \tfrac{9}{2}L)h^2 < 1$ for convergence. We could also examine more elaborate conditions involving both h and h^2, or even take into account that the Lipshcitz constants for f restricted to $[t_0,t_1]$ and $[t_1,t_2]$ could be smaller than L. In any event, even with the least complicated condition it is straightforward to verify that in some situations, depending on the sizes of L and $b - a$, the condition obtained by this partition method is an improvement over the estimate given in Theorems 2.1.3 and 2.1.4 for existence of a solution of (5.1) and the convergence of the sequence of iterates to it.

This section has presented the key ideas in the extension of the method of Chapter 2 based on a partition of the interval $[a,b]$ into subintervals. We now state and prove the three lemmas that were needed in this simplest case, and which will also be needed in the general case discussed in the next section.

Lemma 5.1.2. *Let $x^{[k]}$ be a sequence in a normed vector space $(V, |\cdot|)$. If there exist a number $c < 1$ and an index $N \in \mathbb{N}$ such that*

$$|x^{[k+1]} - x^{[k]}| \leqslant c|x^{[k]} - x^{[k-1]}| \quad \text{for all} \quad k \geqslant N$$

then the sequence $x^{[k]}$ is a Cauchy sequence.

Proof. Applying the condition $|x^{[k+1]} - x^{[k]}| \leqslant c|x^{[k]} - x^{[k-1]}|$ a total of k times we have

$$|x^{[N+k]} - x^{[N+k-1]}| \leqslant c^k|x^{[N]} - x^{[N-1]}| \qquad (k \geqslant 1)$$

so that for any $m > n$ by applying triangle inequality we get

$$|x^{[N+m]} - x^{[N+n]}| \leqslant |x^{[N+m]} - x^{[N+m-1]}| + \cdots + |x^{[N+n+1]} - x^{[N+n]}|$$

$$\leqslant (c^m + c^{m-1} + \cdots + c^{n+1})|x^{[N]} - x^{[N-1]}|$$

$$\leqslant (c^{n+1} + c^{n+2} + \cdots)|x^{[N]} - x^{[N-1]}|$$

$$= \left(\frac{c^{n+1}}{1-c}\right)|x^{[N]} - x^{[N-1]}|$$

$$= \left(\frac{c^n}{1-c}\right)|x^{[N]} - x^{[N-1]}|$$

which can be made arbitrarily small by choosing n sufficiently large. $\square$

Lemma 5.1.3. *Suppose the interval $[a,b]$ has been partitioned into n subintervals of equal length $h = (b - a)/n$ by partition points $a = t_0 < t_1 < \cdots < t_{n-1} < t_n = b$. With the notation*

$$\mathbf{y}_j(t) = \begin{pmatrix} y_j(t) \\ u_j(t) \end{pmatrix} \quad \text{and} \quad f_j^{[r]}(s) = f(s, y_j^{[r]}(s), u_j^{[r]}(s)), \quad r \in \mathbb{Z}^+ \cup \{0\}, \quad j = 1,2,$$

the following estimates hold:

$$\int_{t_{j-1}}^{t_j} |f_j^{[k+1]}(s) - f_j^{[k]}(s)| \, ds \leqslant Lh\|\mathbf{y}^{[k+1]} - \mathbf{y}^{[k]}\|_{max} \tag{5.25}$$

and

$$\int_{t_{j-1}}^{t_j} (t_j - s)|f_j^{[k+1]}(s) - f_j^{[k]}(s)| \, ds \leqslant \tfrac{1}{2} Lh^2 \|\mathbf{y}^{[k+1]} - \mathbf{y}^{[k]}\|_{max}. \tag{5.26}$$

Proof. For

$$\int_{t_{j-1}}^{t_j} |f_j^{[k+1]}(s) - f_j^{[k]}(s)| \, ds \leqslant \int_{t_{j-1}}^{t_j} L\|\mathbf{y}_j^{[k+1]}(s) - \mathbf{y}_j^{[k]}(s)\|_{sum} \, ds$$

$$\leqslant L\|\mathbf{y}_j^{[k+1]} - \mathbf{y}_j^{[k]}\|_{sup} \int_{t_{j-1}}^{t_j} ds \leqslant Lh\|\mathbf{y}^{[k+1]} - \mathbf{y}^{[k]}\|_{max}$$

and

$$\int_{t_{j-1}}^{t_j} (t_j - s)|f_j^{[k+1]}(s) - f_j^{[k]}(s)| \, ds \leqslant \int_{t_{j-1}}^{t_j} (t_j - s) L\|\mathbf{y}_j^{[k+1]}(s) - \mathbf{y}_j^{[k]}(s)\|_{sum} \, ds$$

$$\leqslant L\|\mathbf{y}_j^{[k+1]} - \mathbf{y}_j^{[k]}\|_{sup} \int_{t_{j-1}}^{t_j} t_j - s \, ds$$

$$\leqslant \tfrac{1}{2} Lh^2 \|\mathbf{y}^{[k+1]} - \mathbf{y}^{[k]}\|_{max}. \quad \square$$

Lemma 5.1.4. *Suppose* $\eta : (-\varepsilon, \varepsilon) \to \mathbb{R}$ *is continuous and that* η' *exists on* $(-\varepsilon, 0) \cup (0, \varepsilon)$. *Suppose* $g :$ $(-\varepsilon, \varepsilon) \to \mathbb{R}$ *is continuous and that* $g = \eta'$ *on* $(-\varepsilon, 0) \cup (0, \varepsilon)$. *Then* η' *exists and is continuous on* $(-\varepsilon, \varepsilon)$.

Proof. Since η is continuous at 0 we have the existence and value of the limit

$$\lim_{h \to 0} \eta(h) = \eta(0).$$

This fact and the hypothesis that η' exists on $(-\varepsilon, 0) \cup (0, \varepsilon)$ imply that l'Hôpital's Rule applies, and with the remaining hypotheses gives the existence and value of the limit

$$\eta'(0) = \lim_{h \to 0} \frac{\eta(h) - \eta(0)}{h - 0} = \lim_{h \to 0} \eta'(h) = \lim_{h \to 0} g(h) = g(0).$$

But then η' exists and is equal to the continuous function g on $(-\varepsilon, \varepsilon)$, giving the result. $\square$

5.2 The General Case

If $\eta(t)$ is a solution of the original boundary value problem (5.1) and the interval $[a, b]$ is subdivided into n subintervals of equal length h by means of a partition

$$a = t_0 < t_1 < t_2 < \cdots < t_n = b$$

then setting $\beta_j = \eta(t_j)$, $j = 1, \ldots, n-1$, we see that n boundary value problems are induced:

$$
\begin{array}{ccc}
y'' = f(t, y, y') & y'' = f(t, y, y') & y'' = f(t, y, y') \\
y(t_0) = \alpha, \ y(t_1) = \beta_1 & y(t_1) = \beta_1, \ y(t_2) = \beta_2 & y(t_{n-1}) = \beta_{n-1}, \ y(t_n) = \beta.
\end{array}
$$

Setting $\gamma_j = \eta'(t_{j-1})$, $j = 1, \ldots, n$, or computing them by means of an appropriate implementation of (5.2b), their solutions are solutions of the respective initial value problems

$$
\begin{array}{ccc}
y'' = f(t, y, y') & y'' = f(t, y, y') & y'' = f(t, y, y') \\
y(t_0) = \alpha, \ y'(t_0) = \gamma_1 & y(t_1) = \beta_1, \ y'(t_1) = \gamma_2 & y(t_{n-1}) = \beta_{n-1}, \ y'(t_{n-1}) = \gamma_n.
\end{array}
$$

As in the previous section we denote the solutions to these problems by $y_j(t)$ with derivatives $u_j(t) := y_j'(t)$, $j = 1, 2, 3$. See Figure (5.2)..
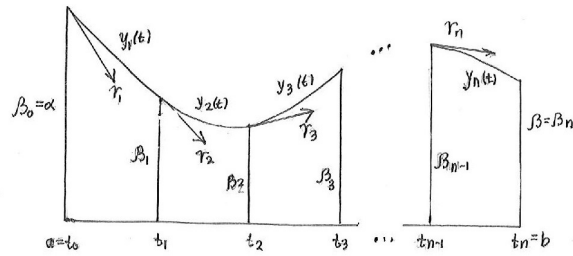
To make the presentation cleaner and easier to read we continue and expand the shorthand notation of the previous section (for relevant choices of j):

$$f_j(s) = f(s, y_j(s), u_j(s)) \qquad f_j^{[k]}(s) = f(s, y_j^{[k]}(s), u_j^{[k]}(s))$$

$$\mathbf{y}_j(s) = \begin{pmatrix} u_j(s) \\ y_j(s) \end{pmatrix} \qquad \mathbf{y}_j^{[k]}(s) = \begin{pmatrix} u_j^{[k]}(s) \\ y_j^{[k]}(s) \end{pmatrix}$$

(5.27)

$$I_j(s) = \int_{t_{j-1}}^{t_j} f_j(s)\, ds \qquad I_j^{[k]}(s) = \int_{t_{j-1}}^{t_j} f_j^{[k]}(s)\, ds$$

$$J_j(s) = \int_{t_{j-1}}^{t_j} (t_j - s) f_j(s)\, ds \qquad J_j^{[k]}(s) = \int_{t_{j-1}}^{t_j} (t_j - s) f_j^{[k]}(s)\, ds.$$

The idea for generating a sequence of successive approximations of $\eta(t)$ is to update the estimates of the functions $\mathbf{y}_j(t)$ using (5.2a) in the form

$$\mathbf{y}_j(t) = \begin{pmatrix} \beta_{j-1} \\ \gamma_j \end{pmatrix} + \int_{t_{j-1}}^{t_j} \begin{pmatrix} u_j(s) \\ f_j(s) \end{pmatrix} ds \tag{5.28}$$

and then update γ_j and β_j using (5.4) and (5.3) in the forms

$$\gamma_j = \gamma_{j-1} + I_{j-1} \qquad \text{and} \qquad \beta_{j-1} = \beta_j - h\gamma_j - J_j \tag{5.29}$$

(starting with $j = 1$ and working our way up to $j = n$ for the γ_j and in the reverse order with the β_j, with the convention that $\beta_n = \beta$), except that on the first step we update γ_1 using instead (5.2b) in the form

$$\gamma_1 = \frac{1}{h}[\beta_1 - \alpha - J_1] \tag{5.30}$$

and there is no β_0. But as in the case $n = 2$ we will not be able to make the estimates that we need to show convergence if we update β_1 on this basis. We will address this problem in a manner directly analogous to what was done in the simpler case. Before we do, however, note that the updates on the β_j come from "the right," i.e., values of β_r with $r > j$, hence ultimately tying into β at each pass through the recursion, while the updates on the γ_j come from "the left," i.e., values of γ_r with $r < j$, hence ultimately tying into α at each pass through the recursion.

To obtain a useful formula on which to base the successive approximations of β_1, we begin by using the second formula in (5.29) $n - 1$ times:

$$\begin{aligned} \beta_1 &= \beta_2 - (h\gamma_2 + J_2) \\ &= \beta_3 - (h\gamma_3 + J_3) - (h\gamma_2 + J_2) \\ &= \beta_4 - (h\gamma_4 + J_4) - (h\gamma_3 + J_3) - (h\gamma_2 + J_2) \\ &\;\;\vdots \\ &= \beta - (h\gamma_n + J_n) - \cdots - (h\gamma_2 + J_2) \\ &= \beta - h(\gamma_2 + \cdots + \gamma_n) - (J_2 + \cdots + J_n). \end{aligned}$$

But by repeated application of the first equation in (5.29) and use of (5.30) on the last step

$$\gamma_2 + \cdots + \gamma_{n-2} + \gamma_{n-1} + \gamma_n$$

$$= \gamma_2 + \cdots + \gamma_{n-2} + 2\gamma_{n-1} + I_{n-1}$$

$$= \gamma_2 + \cdots + 3\gamma_{n-2} + 2I_{n-2} + I_{n-1}$$

$$\vdots$$

$$= (n-1)\gamma_2 + (n-2)I_2 + \cdots + 3I_{n-3} + 2I_{n-2} + I_{n-1}$$

$$= (n-1)\gamma_1 + (n-1)I_1 + (n-2)I_2 + \cdots + 3I_{n-3} + 2I_{n-2} + I_{n-1}$$

$$= \frac{n-1}{h}[\beta_1 - \alpha - J_1] + (n-1)I_1 + (n-2)I_2 + \cdots + 3I_{n-3} + 2I_{n-2} + I_{n-1}.$$

Inserting this expression into the previous display and solving the resulting equation for β_1 yields the formula

$$\beta_1 = \frac{1}{n}\left[\beta + (n-1)\alpha + (n-1)J_1 - \sum_{r=2}^{n} J_r - h\sum_{r=1}^{n-1}(n-r)I_r\right]. \tag{5.31}$$

Once an initialization has been chosen, an iteration procedure based on (5.28), (5.29), (5.30), and (5.31) is, with the convention $\beta_0 = \alpha$ and $\beta_n = \beta$, the shorthand notation introduced above, and order of evaluation in the order listed,

$$\mathbf{y}_j^{[k+1]}(t) = \begin{pmatrix} \beta_{j-1}^{[k]} \\ \gamma_j^{[k]} \end{pmatrix} + \int_{t_{j-1}}^{t_j} \begin{pmatrix} u_j^{[k]}(s) \\ f_j^{[k]}(s) \end{pmatrix} ds \qquad j = 1, \cdots, n \tag{5.32a}$$

$$\beta_1^{[k+1]} = \frac{1}{n}\left[\beta + (n-1)\alpha + (n-1)J_1^{[k+1]} - \sum_{r=2}^{n} J_r^{[k+1]} - h\sum_{r=1}^{n-1}(n-r)I_r^{[k+1]}\right] \tag{5.32b}$$

$$\gamma_1^{[k+1]} = \frac{1}{h}[\beta_1^{[k+1]} - \alpha - J_1^{[k+1]}] \tag{5.32c}$$

$$\gamma_j^{[k+1]} = \gamma_{j-1}^{[k+1]} + I_{j-1}^{[k+1]} \qquad j = 2, \cdots, n \tag{5.32d}$$

$$\beta_{j-1}^{[k+1]} = \beta_j^{[k+1]} - h\gamma_j^{[k+1]} - J_j^{[k+1]} \qquad j = n, n-1, \ldots, 4, 3 \tag{5.32e}$$

Theorem 5.2.1. *Suppose the function $f(t, y, u)$ from $[a, b] \times \mathbb{R}^2$ into $\mathbb{R}$ is continuous and Lipschitz in $\mathbf{y} = (y, u)$ with Lipschitz constant L. If there exists an integer $n \geqslant 1$ such that for the subdivision of $[a, b]$ into n*

subintervals of equal length h by the partition

$$a = t_0 < t_1 < \cdots < t_{n-1} < t_n = b$$

the inequality

$$\frac{1}{2n}[(n^3 + n^2 + n + 2)L + 2](b - a) < 1$$

holds if $h = (b-a)/n \leqslant 1$ or the inequality

$$\frac{1}{2n^2}[(n^3 + n^2 + n + 2)L + 2](b - a)^2 < 1$$

holds if $h = (b-a)/n \geqslant 1$, then there exists a solution of the two-point boundary value problem (5.1). *More-over, in the language of the notation of Figure B and display* (5.27), *for any initial choice of the functions $\mathbf{y}_j(t) = (y_j(t), u_j(t))$, $1 \leqslant j \leqslant n$, the constants γ_j, $1 \leqslant j \leqslant n$, and the constants β_j, $1 \leqslant j \leqslant n-1$, the sequence of successive approximations defined by* (5.32) *converges to such a solution.*

Proof. As was done in the simple case $n = 2$ we will show that the sequence

$$\mathbf{y}(t) = (\mathbf{y}_1(t), \ldots, \mathbf{y}_n(t)) \in C([t_0, t_1], \mathbb{R}^2) \times C([t_1, t_2], \mathbb{R}^2) \times \cdots \times C([t_{n-1}, t_n], \mathbb{R}^2)$$

is a Cauchy sequence by means of Lemma 5.1.4, where we place the supremum norm on each function space, with respect to absolute value on $\mathbb{R}$ and the sum norm on $\mathbb{R}^2$, and the maximum norm on their product. Precisely the same reasoning as was given in the paragraph that follows display (5.15) shows that this is sufficient to show that the successive approximations converge to functions on the individual subintervals which when concatenated form a solution of (5.1), appealing to Lemma 5.1.4 for existence of the second derivative at the partition points.

By Lemma 5.1.3

$$
\begin{aligned}
|I_r^{[k+1]} - I_r^{[k]}| &= \left| \int_{t_{r-1}}^{t_r} f_r^{[k+1]}(s)\,ds - \int_{t_{r-1}}^{t_r} f_r^{[k]}(s)\,ds \right| \\
&\leqslant \int_{t_{r-1}}^{t_r} |f_r^{[k+1]}(s) - f_r^{[k]}(s)|\,ds \\
&\leqslant Lh \|\mathbf{y}^{[k+1]} - \mathbf{y}^{[k]}\|_{\max}
\end{aligned}
\tag{5.33}
$$

and

$$
\begin{aligned}
|J_r^{[k+1]} - J_r^{[k]}| &= \left| \int_{t_{r-1}}^{t_r} (t_r - s)f_r^{[k+1]}(s)\,ds - \int_{t_{r-1}}^{t_r} (t_r - s)f_r^{[k]}(s)\,ds \right| \\
&\leqslant \int_{t_{r-1}}^{t_r} (t_r - s)|f_r^{[k+1]}(s) - f_r^{[k]}(s)|\,ds \\
&\leqslant \tfrac{1}{2}Lh^2 \|\mathbf{y}^{[k+1]} - \mathbf{y}^{[k]}\|_{\max}.
\end{aligned}
\tag{5.34}
$$

Then from (5.32b) we have

$$|\beta_1^{[k+1]} - \beta_1^{[k]}| \leqslant \frac{n-1}{n}\left|J_1^{[k+1]} - J_1^{[k]}\right| + \sum_{r=2}^{n}\left|J_r^{[k+1]} - J_r^{[k]}\right| + h\sum_{r=1}^{n-1}(n-r)\left|I_r^{[k+1]} - I_r^{[k]}\right|$$

$$\leqslant \left[\left(\frac{n-1}{n}\right)\frac{1}{2}h^2 + \sum_{r=2}^{n}\frac{1}{2}h^2 + h\sum_{r=1}^{n-1}(n-r)h\right]L||\mathbf{y}^{[k+1]} - \mathbf{y}^{[k]}||_{\max}$$

$$= \widehat{B}_1 h^2 L||\mathbf{y}^{[k+1]} - \mathbf{y}^{[k]}||_{\max},$$

where $\widehat{B}_1 = \frac{n-1}{2}\left[\frac{1}{n} + 1 + n\right]$.

From (5.32c)

$$|\gamma_1^{[k+1]} - \gamma_1^{[k]}| \leqslant \frac{1}{h}|\beta_1^{[k+1]} - \beta_1^{[k]}| + \frac{1}{h}|J_1^{[k+1]} - J_1^{[k]}|$$

$$\leqslant [\widehat{B}_1 hL + \frac{1}{2}hL]||\mathbf{y}^{[k+1]} - \mathbf{y}^{[k]}||_{\max}$$

$$= \widehat{\Gamma}_1 hL||\mathbf{y}^{[k+1]} - \mathbf{y}^{[k]}||_{\max}$$

and from repeated application of (5.32d), starting from $j = 2$ up through $j = n$,

$$|\gamma_j^{[k+1]} - \gamma_j^{[k]}| \leqslant |\gamma_{j-1}^{[k+1]} - \gamma_{j-1}^{[k]}| + |I_{j-1}^{[k+1]} - I_{j-1}^{[k]}|$$

$$\leqslant [\widehat{\Gamma}_{j-1} hL + hL]||\mathbf{y}^{[k+1]} - \mathbf{y}^{[k]}||_{\max}$$

$$= \widehat{\Gamma}_j hL||\mathbf{y}^{[k+1]} - \mathbf{y}^{[k]}||_{\max}$$

with $\widehat{\Gamma}_j = \widehat{B}_1 + \frac{2j-1}{2}$, which is in fact valid for $1 \leqslant j \leqslant n$.

From repeated application of (5.32e), starting from $j = n-1$ (with the convention that $\beta_n = \beta$) and down through $j = 2$,

$$|\beta_j^{[k+1]} - \beta_j^{[k]}| \leqslant |\beta_{j+1}^{[k+1]} - \beta_{j+1}^{[k]}| + h|\gamma_{j+1}^{[k+1]} - \gamma_{j+1}^{[k]}| + |J_{j+1}^{[k+1]} - J_{j+1}^{[k]}|$$

$$\leqslant [\widehat{B}_{j+1} h^2 L + \widehat{\Gamma}_{j+1} h^2 L + \frac{1}{2}h^2 L]||\mathbf{y}^{[k+1]} - \mathbf{y}^{[k]}||_{\max}$$

$$= \widehat{B}_j h^2 L||\mathbf{y}^{[k+1]} - \mathbf{y}^{[k]}||_{\max},$$

and $\widehat{B}_{n-r} = r\widehat{B}_1 + rn - \frac{r(r-1)}{2}$, hence $\widehat{B}_j = (n-j)\widehat{B}_1 + n(n-j) - \frac{(n-j)(n-j-1)}{2}$, $2 \leqslant j \leqslant n-1$.

Using these estimates we find that, setting $h^* = \max\{h, h^2\}$, for any $j \in \{2, \ldots, n\}$, for any $t \in [t_{j-1}, t_j]$,

$$
\begin{aligned}
|\mathbf{y}_j^{[k+1]}(t) - \mathbf{y}_j^{[k]}(t)|_{\text{sum}} \\
\leqslant \left| \begin{pmatrix} \beta_{j-1}^{[k]} - \beta_{j-1}^{[k-1]} \\ \gamma_j^{[k]} - \gamma_j^{[k-1]} \end{pmatrix} \right|_{\text{sum}} + \int_{t_{j-1}}^{t_j} \left| \begin{pmatrix} u_j^{[k]}(s) - u_j^{[k-1]}(s) \\ f_j^{[k]}(s) - f_j^{[k-1]}(s) \end{pmatrix} \right|_{\text{sum}} ds \\
\leqslant |\beta_{j-1}^{[k]} - \beta_{j-1}^{[k-1]}| + |\gamma_j^{[k]} - \gamma_j^{[k-1]}| + (1+L) \int_{t_{j-1}}^{t_j} |\mathbf{y}_j^{[k]}(s) - \mathbf{y}_j^{[k-1]}(s)|_{\text{sum}} ds \\
\leqslant [\widehat{B}_{j-1} h^2 L + \widehat{\Gamma}_j hL] \, ||\mathbf{y}^{[k]} - \mathbf{y}^{[k-1]}||_{\text{max}} + (1+L)||\mathbf{y}_j^{[k]} - \mathbf{y}_j^{[k-1]}||_{\text{sup}} h \\
\leqslant [\widehat{B}_{j-1} L + \widehat{\Gamma}_j L + (1+L)]h^* ||\mathbf{y}^{[k]} - \mathbf{y}^{[k-1]}||_{\text{max}} \\
= [(\widehat{B}_{j-1} + \widehat{\Gamma}_j + 1)L + 1]h^* ||\mathbf{y}^{[k]} - \mathbf{y}^{[k-1]}||_{\text{max}} \\
= c_j h^* ||\mathbf{y}^{[k]} - \mathbf{y}^{[k-1]}||_{\text{max}}
\end{aligned}
$$

where, using the expressions above for $\widehat{B}_1$ and $\widehat{\Gamma}_1$,

$$
c_j = \tfrac{1}{2n}[n^4 + (3-j)n^3 + n^2 + (3j - j^2)n + (j-2)]L + 1 \qquad (2 \leqslant j \leqslant n).
$$

Similarly, for all $t \in [t_0, t_1]$, $|\mathbf{y}_1^{[k+1]}(t) - \mathbf{y}_1^{[k]}(t)|_{\text{sum}} \leqslant c_1 h ||\mathbf{y}^{[k]} - \mathbf{y}^{[k-1]}||_{\text{max}}$ for

$$
c_1 = \left[\frac{n^3 + 3n - 1}{2n} \right] L + 1.
$$

Then for all $j \in \{1, \ldots, n\}$,

$$
||\mathbf{y}_j^{[k]} - \mathbf{y}_j^{[k-1]}||_{\text{sup}} \leqslant c_j h^* ||\mathbf{y}^{[k]} - \mathbf{y}^{[k-1]}||_{\text{max}}
$$

and

$$
||\mathbf{y}^{[k+1]} - \mathbf{y}^{[k]}||_{\text{max}} \leqslant \max\{c_1, \ldots, c_n\} h^* ||\mathbf{y}^{[k]} - \mathbf{y}^{[k-1]}||_{\text{max}}. \tag{5.35}
$$

For fixed n, for $j \geqslant 2$, c_j is a quadratic function of j with maximum at $j = \frac{-n^3 + 3n + 1}{2n}$, which is negative for $n \geqslant 2$, so $c_2 > c_j$ for $j \geqslant 3$. Direct comparison shows that $c_2 > c_1$ for all choices of n as well. Thus estimate (5.35) is

$$
||\mathbf{y}^{[k+1]} - \mathbf{y}^{[k]}||_{\text{max}} \leqslant c_2 h^* ||\mathbf{y}^{[k]} - \mathbf{y}^{[k-1]}||_{\text{max}}
$$

and by Lemma 5.1.2 the sequence $\mathbf{y}^{[k]}(t)$ is a Cauchy sequence provided $c_2 h^* < 1$, which, when $h = (b-a)/n \leqslant 1$ is the condition

$$
\frac{1}{2n}[(n^3 + n^2 + n + 2)L + 2](b-a) < 1
$$

and when $h = (b-a)/n \geqslant 1$ is the condition

$$\frac{1}{2n^2}[(n^3 + n^2 + n + 2)L + 2](b-a)^2 < 1. \quad \square$$

5.3 Examples

In this section we illustrate the method by means of several examples.
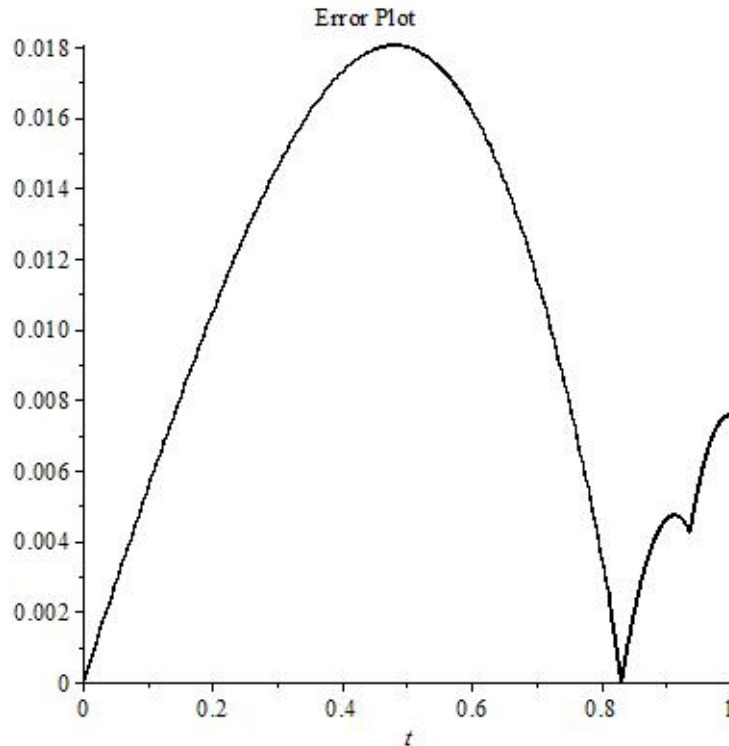
Example 5.3.1. Consider

$$y'' = 2y^3$$
$$y(0) = -\frac{1}{2} \qquad\qquad (5.36)$$
$$y(1) = -1$$

The exact solution is $y(t) = \frac{1}{t-2}$. Without loss of generality we divided the interval $[0,1]$ into the 3-subintervals not necessary of the same lengths.

The reason we pick different length because the Lipschitz constant for the entire interval is same as the last subinterval or I should say If we look at subintervals then Lipschitz constant will be increasing for subintervals from left to right. Due to this fact then we can have have length of subintervals decreasing from left to right. However, in general we do not have a nice algorithm to do that so to be safe we need to make each subinterval less than $\frac{1}{1+\frac{3}{2}L}$ which is provided by theorem (2.1.4). For this problem the Lipschitz constant is $L = 6$ and in order to be in the safe side we need interval's length to be less than $\frac{1}{1+\frac{3}{2}(6)} = \frac{1}{10}$. So we introduce the following subintervals

$$[0, \frac{13}{16}], \ [\frac{13}{16}, \frac{15}{16}], \ [\frac{15}{16}, 1]$$

After 4 iterations we will have a maximum error of $E = 0.018$ and error plot will be

Error Plot

Example 5.3.2. Consider again the problem (5.37) but this time with different boundary conditions

$$y'' = 2y^3$$

$$y(0) = -\frac{10}{11} \tag{5.37}$$

$$y(1) = -10$$

The exact solution is $y(t) = \frac{1}{t-1.1}$. The Lipschitz constant is $L = 600$ and in order to be in the safe side we need interval's length to be less than $\frac{1}{1+\frac{3}{2}(600)} = \frac{1}{901}$. This problem can be solved by more sophisticated software such as "MATLAB". The softwares "MAPLE" and "MATHEMATICA" are not recommended here since they are mostly symbolically software.

Example 5.3.3. Consider

$$y'' = -e^{-2y}$$

$$y(0) = 0 \tag{5.38}$$

$$y(1) = \ln(2)$$

The exact solution is $y(t) = \ln(1+t)$. Without loss of generality we divided the interval $[0,1]$ into the 3-subintervals not necessary of the same size. For this problem the Lipschitz constant is $L = \frac{1}{2}$ and in order to be in the safe side we need interval's length to be less than $\frac{1}{1+\frac{3}{2}(\frac{1}{2})} = \frac{4}{7}$. So we introduce the following

subintervals

$$[0, \frac{3}{4}], \ [\frac{3}{4}, \frac{7}{8}], \ [\frac{7}{8}, 1]$$

After 4 iterations we will have a maximum error of $E = 0.028$ and error plot will be

Error Plot