On Continuous Approximate Solution Of Ordinary Differential Equations

De Ting Wu dtwu@morehouse.edu
Department of Mathematics
Morehouse College
Atlanta, GA. USA

Abstract

In this work the problem of continuous approximate solution of the ordinary differential equations will be investigated. An approach to construct the continuous approximate solution, which is based on the discrete approximate solution and the spline interpolation, will be provided. The existence and uniqueness of such continuous approximate solution will be pointed out. Its error will be estimated and its convergence will be considered. Finally, with the aid of modern PC and mathematical software, three practical computer approaches to perform above construction will be offered.
AMS Classification Subject: 34A45, 65L05, 65Y99

Key Words: Continuous Approximate Solution, Discrete Approximate Solution, Cubic Spline Interpolant.

1. INTRODUCTION

1.1 Presentation of Problem

Differential equations are often used to model, understand and predict the dynamic systems in the real world. The use of differential equations makes available to us the full power of Calculus. Modeling by differential equations greatly expands the list of possible applications of Mathematics.

Unfortunately, a wide majority of interesting differential equations have no closed form. i.e. the solution can't be expressed explicitly in terms of elementary functions such as polynomial,exponential, logarithmic or trigonometric functions even if it can be shown that a solution of the differential equation exists. Furthermore, in many case the explicit solution does exist, but the evaluation of the function may be difficult. Thus, we have to be content with an approximation of the solution of differential equations and the approximate methods for differential equations developed.

Generally, the approximate methods fall into two categories:
(1)Discrete approximate methods which produce a table of approximation of solution values corresponding to points of independent variable. This kind of methods provides quatitative information about solution even if we can not find the formula of the solution. There's also advantage that most of work can be done by machines. However, its disadvantage is that we obtain only approximation, not precise solution, and not a function.

(2)Continuous approximate methods on which much less work has been done. Although theoretically any discrete approximate methods can be converted to continuous approximate method by interpolating, but there remains a lot of problems not answered thoroughly, such as error, convergence,stability and computer approach . In this work, we attemp to solve some of these problems.

1.2 Description of Problem

In this work we'll study the problem of continuous approximate solution of initial value problem of ordinary differential equation:

$$y' = f(x,y) \quad \text{and} \quad y(a) = y_0 \quad x \text{ in } [a,b] \qquad (1.2.1)$$

Let f(x,y) be continuous and satisfy Lipschitz condition on D where D={ (x,y): $a \le x \le b \quad -\infty < y < \infty$ } then (1.2.1) has unique solution:

$$y = y(x) \quad x \text{ in } [a,b]. \qquad (1.2.2)$$

Suppose we have a mesh of [a,b]: $a=x_0 < x_1 < ...... < x_n=b$

then its exact solution values at points in mesh are $y(x_i) = y_i \quad i = 0, 1, ..., n$.

If we use certain discrete approximate method of (1.2.1) to produce its discrete approximate solution:

$$\{ (x_i, w_i) : i = 0, 1, ...,n \} \qquad (1.2.3)$$

where w_i is the approximation of y_i with error $\left|y_i - w_i\right| = O\left(h_i\right)^p$.

In this work, we assume (1.2.1) is a scalar equation, but most theoretical and numerical consideration can be carried over to vector form -- the system of 1st order equations. And, we assume (1.2.1) satisfies stronger differentiation conditions as needed in theoretical analysis later.

1.3 Natural Cubic Spline Interpolant

In this work the natural cubic spline interpolant will be used to construct the continuous approximate solution of (1.2.1) which is defined as follows.

Definition of Natural Cubic Spline Interpolant

For a set of data { (x_i, w_i) i = 1, 2, ...,n } its natural cubic spline interpolant is a piece-wise cubic polynomial s(x), for x in $[x_i, x_{i+1}]$ $s(x) = s_i(x)$

where $s_i(x) = a_i \cdot \left(x - x_i\right)^3 + b_i \cdot \left(x - x_i\right)^2 + c_i \cdot \left(x - x_i\right) + d_i \quad i = 0, 1, 2, ...,(n-1) \qquad (1.3.1)$

which meet following conditions:

agreeing with data: $s_i\left(x_i\right) = w_i \quad i = 0, 1, 2,......,(n-1)$ and $s_{n-1}\left(x_n\right) = w_n \qquad (1.3.2.a)$

function values of adjacent 2 pieces are equal at joint points:

$$s_i\left(x_{i+1}\right) = s_{i+1}\left(x_{i+1}\right) \qquad i = 0, 1, 2, ...,(n-2) \qquad (1.3.2.b)$$

1st derivative values of 2 adjacent pieces are equal at joint points:

$$s_i'(x_{i+1}) = s_{i+1}'(x_{i+1}) \qquad i = 0, 1, 2, ...,(n-2) \qquad (1.3.2.c)$$

2nd derivative values of 2 adjacent pieces are equal at joint points:

$$s_i''\left(x_{i+1}\right) = s_{i+1}''\left(x_{i+1}\right) \qquad i = 0, 1, 2, ...,(n-2) \qquad (1.3.2.d)$$

Boundary condition of natural spline: $s''\left(x_0\right) = s''\left(x_n\right) = 0 \qquad (1.3.2.e)$

Remark:

1. The natural cubic spline interpolant s(x) is in $C^2([a,b])$.
2. Generally, the cubic spline interpolant is not unique for a set of data. In order to get a unique cubic spline interpolant we need some boundary conditions. In this work we adopt the natural conditions, i.e. the 2nd derivative at two endpoints of interval are equal zero. There're several boundary conditions for cubic spline interpolant, but just cause a slight difference in the following numerical and theoretical consideration and results.

1.4 Computer Approach and Mathematical Software

One of advantage of discrete approximate methods is that it can be performed by computation machines and the machines can do most work of this kind of methods for us. In a long time it is difficult to perform a continuous approximate method by machines.

Now, modern computer technology makes it possible to perform the continuous approximate methods by PC and mathematical software. In this work we'll provide three computer approaches to construct the continuous approximate solution of (1.2.1).

These approach is based on PC and Mathcad 13. Mathcad 13 is a CAS and one of popular mathematical softwares in the world. We'll use its program function and routines to accomplish the construction of continuous approximate solution of (1.2.1)

2. Continuous Approximate Solution

In this part we'll provide an approach to construt the continuous approximate solution of (1.2.1) and discuss its existence and uniqueness. Also, the error will be estimated and the convergence will be considered.

2.1 Approach to Construct the Continuous Approximate Solution

For (1.2.1) and a simple mesh: $a = x_0 < x_1 < x_2 < \ldots\ldots < x_n = b$,

we can obtain a set of data { (x_i, w_i) i = 0,1,2, ...,n} where w_i is approximate solution values produced by certain discrete approximate method.

Then, we form a natural cubic spline interpolant (1.3.1) for the set of data { (x_i, w_i) i = 0,1,2, ...,n}

"To form a natural cubic spline interpolant" means "To determine {a_i, b_i, c_i, d_i, i = 0,1, ...,n-1} in (1.3.1)".

Let $h_i = x_{i+1} - x_i$ i = 0, 1,......, n-1 , then

From (1.3.2.a) $d_i = w_i$ i = 0,1,......,n-1 and

$$a_{n-1}\left(h_{n-1}\right)^3 + b_{n-1}\left(h_{n-1}\right)^2 + c_{n-1}h_{n-1} + d_{n-1} = w_n$$

From (1.3.2.b) $a_i\cdot\left(h_i\right)^3 + b_i\cdot\left(h_i\right)^2 + c_i h_i + d_i = d_{i+1}$ i = 0, 1, ..., n-2 (2.1.0)

From (1.3.2.c) $3a_i\left(h_i\right)^2 + 2b_i h_i + c_i = c_{i+1}$ i = 0, 1, ..., n-2

From (1.3.2.d) $6a_i h_i + 2b_i = 2b_{i+1}$ i = 0, 1, ..., n-2

From (1.3.2.e) $2b_0 = 0$ and $6a_{n-1}h_{n-1} + 2b_{n-1} = 0$

This is a system of linear equations in 4n unknowns $\{a_i, b_i, c_i, d_i, i = 0, 1, ..., n-1\}$. In order to make it easy to solve, program and analyze theoritically we simlify above system as follows.

Let $s''(x_i) = u_i$ i = 0, 1, ...,n with $s''(x_0) = u_0 = 0$ and $s''(x_n) = u_n = 0$

then we have:

$$h_{i-1}\cdot u_{i-1} + 2\cdot(h_{i-1} + h_i)\cdot u_i + h_i\cdot u_{i+1} = 6\cdot\left(\frac{w_{i+1} - w_i}{h_i} - \frac{w_i - w_{i-1}}{h_{i-1}}\right) \qquad i = 1,2,......,n-1 \qquad (2.1.1)$$

also

$$a_i = \frac{u_{i+1} - u_i}{6h_i} \qquad i = 0,1,......,n-1 \qquad (2.1.2)$$

$$b_i = \frac{u_i}{2} \qquad i = 0,1,......,n-1 \qquad (2.1.3)$$

$$c_i = \frac{w_{i+1} - w_i}{h_i} - \frac{u_{i+1} + 2\cdot u_i}{6} h_i \qquad i = 0,1,......,n-1 \qquad (2.1.4)$$

$$d_i = w_i \qquad i = 0,1,......,n-1 \qquad (2.1.5)$$

Later we need the matrix form of (2.1.1) which is TU=W
Where

$$T=\begin{bmatrix} 2(h_1+h_2) & h_2 & 0 & . & . & . & . & . \\ 0 & h_2 & 2(h_2+h_3) & h_3 & 0 & . & . & . \\ ..... & .... & . & .... & . & . & .... & . \\ .... & 0 & h_i & 2(h_i+h_{i+1}) & h_{i+1} & . & 0 & .... \\ . & . & . & .... & . & . & . & . \\ . & . & . & . & . & . & . & . \\ . & . & . & . & .... & . & . & . \\ ..... & . & .... & . & 0 & . & h_{n-1} & 2(h_{n-1}+h_n) \end{bmatrix} \quad (2.1.6)$$

$$U=\begin{pmatrix} u_1 \\ u_2 \\ . \\ u_i \\ . \\ . \\ u_{n-1} \end{pmatrix} \quad (2.1.7) \qquad W=6\begin{pmatrix} \frac{w_2-w_1}{h_1}-\frac{w_1-w_0}{h_0} \\ \frac{w_3-w_2}{h_2}-\frac{w_2-w_1}{h_1} \\ . \\ \frac{w_{i+1}-w_i}{h_i}-\frac{w_i-w_{i-1}}{h_{i-1}} \\ . \\ . \\ \frac{w_n-w_{n-1}}{h_{n-1}}-\frac{w_{n-1}-w_{n-2}}{h_{n-2}} \end{pmatrix} \quad (2.1.8)$$

Summary:

The approach to construct a continuous approximate solution of (1.2.1) as follows:

1.Use a discrete approximate method to find an approximate solution { (x_i, w_i): i=0,1,...,n}

2.Solve system (2.1.1) for { u_i : i = 1, 2,....,n-1) with $u_0 = 0$ and $u_n = 0$

3.Find { a_i, b_i, c_i, d_i, i = 0,1,......,n-1 } by (2.1.2)--(2.1.5)

4 Form the natural cubic spline interpolant by (1.3.1) that is our desired continuous approximate solution of (1.2.1)

2.2 Existence and Uniqueness of Continuous Approximate Solution

The existence and uniqueness of such continuous approximate solution is stated in the following theorem.

Theorem 2.2.1 *For initial value problem (1.2.1) with discrete approximate solution (1.2.3) a unique continuous approximate solution, determined by the natural cubic spline interpolant (1.3.1), exists.*

Proof: The continuous approximate solution is the natural cubis spline interpolant (1.3.1) which is determined by its coeffients { (a_i, b_i, c_i, d_i) i = 0,1,...,n-1 }. These coefficients are solution of (2.1.1)--(2.1.5). (2.1.1) is a system of linear equations of (n-1)X(n-1) and the matrix T (2.1.6) is strictly diagonally dominant and nonsigular, so (2.1.1) has unique solution. Thus, The natural cubic spline interpolant exists and is unique.

2.3 Error of Continuous Approximate Solution

In this part, 1st we prove a lemma and then estimate the the error of such continuous approximate solution.

Lemma 2.3.1 *For a system of linear equations (2.1.1) TU = W then* $||U|| \le \frac{||W||}{2h}$ *(2.3.1)*

where || U || and || W || are max norm of U and W and h = min { h_i*, i = 0,1,......,n-1 },*

Proof: Let $||U||=|u_i|$, then from i th equation of (1.2.1) we have:

$$| u_{i-1} h_{i-1} + 2u_i\left(h_{i-1} + h_i\right) + u_{i+1} h_i | = | w_i |$$

$$| u_i | \left| \frac{u_{i-1}}{u_i} h_{i-1} + 2\left(h_{i-1} + h_i\right) + \frac{u_{i+1}}{u_i} h_i \right| = | w_i |$$

Then, $\left|u_i\right|\left(h_{i-1} + h_i\right) \le \left|w_i\right|$ i.e $||U|| \le \frac{\left|w_i\right|}{h_{i-1} + h_i} \le \frac{||W||}{2h}$

Theorem 2.3.2 *For initial value problem (1.2.1) wlth discrete approximate solution (1.2.3) of order p if its continuous approximate solution s(x), determined by the natural cubic spline interpolant (1.3.1), then error with exact solution y(x):*

$$||y(x) - s(x)|| \le K\cdot\left|\left|y^{(4)}\right|\right|\cdot H^4 + H^p\left[c_3\cdot\left(\frac{H}{h}\right)^3 + c_2\cdot\left(\frac{H}{h}\right)^2 + c_1\cdot\frac{H}{h} + c_0\right] \qquad (2.3.2)$$

Proof: | y(x) - s(x) | is continuous on [a,b], it has max and min on [a,b]. Assume at x in [x_i, x_{i+1}] , | y(x) - s_i(x) | is max, then || y(x) - s(x) || = | y(x) - s_i(x) | .

and we have: $$\left|y(x) - s_i(x)\right| \le \left|y(x) - g_i(x)\right| + \left|g_i(x) - s_i(x)\right| \qquad (2.3.3)$$

where $g_i(x) = a1_i\left(x - x_i\right)^3 + b1_i\left(x - x_i\right)^2 + c1_i\left(x - x_i\right) + d1_i$ The natural cubic spline interpolant for data set { (x_i, y_i) : i = 0,1,....,n } where y_i is the exact solution value of y(x) at x_i .

Let H = max{h_i : i = 0,1,......,n-1} and h = min{ h_i : i = 0,1, ...,n-1 } then, Assume y(x) of (1.2.1) in $C^{(4)}[a,b]$ with $\left\|y^{(4)}\right\| \le M$ then the first term in (2.3.3) (see [1])

$$\left\|y(x) - g_i(x)\right\| \le K\cdot\left\|y^{(4)}\right\|\cdot H^4 \tag{2.3.4}$$

Now, let's estimate the second term in (2.3.3)

$$\left|g_i\cdot(x) - s_i\cdot(x)\right| = \left|\begin{array}{l}\left[a1_i\cdot\left(x-x_i\right)^3 + b1_i\cdot\left(x-x_i\right)^2 + c1_i\cdot\left(x-x_i\right) + d1_i\right]\ldots \\ +\left[-a_i\cdot\left(x-x_i\right)^3 - b_i\cdot\left(x-x_i\right)^2 - c_i\cdot\left(x-x_i\right) - d_i\right]\end{array}\right|$$

$$\left|g_i\cdot(x) - s_i\cdot(x)\right| \le \left|a1_i - a_i\right|\cdot H^3 + \left|b1_i - b_i\right|\cdot H^2 + \left|c1_i - c_i\right|\cdot H + \left|d1_i - d_i\right| \tag{2.3.5}$$

where from (2.1.2) $\left|a1_i - a_i\right| = \left|\dfrac{u1_{i+1} - u1_i}{6h_i} - \dfrac{u_{i+1} - u_i}{6h_i}\right|$ $\quad\left|a1_i - a_i\right| \le \dfrac{\left|u1_{i+1} - u_{i+1}\right| + \left|u1_i - u_i\right|}{6h_i}$

from (2.1.3) $\left|b1_i - b_i\right| = \left|\dfrac{u1_i}{2} - \dfrac{u_i}{2}\right|$ $\quad\left|b1_i - b_i\right| \le \dfrac{\left|u1_i - u_i\right|}{2}$

from (2.1.4) $\left|c1_i - c_i\right| = \left|\left(\dfrac{y_{i+1} - y_i}{h_i} - \dfrac{u1_{i+1} + 2u1_i}{6}\cdot h_i\right) - \left(\dfrac{w_{i+1} - w_i}{h_i} - \dfrac{u_{i+1} + 2u_i}{6}\cdot h_i\right)\right|$

$$\left|c1_i - c_i\right| \le \frac{\left|y_{i+1} - w_{i+1}\right| + \left|y_i - w_i\right|}{h} + \frac{\left|u1_{i+1} - u_{i+1}\right| + 2\left|u1_i - u_i\right|}{6}\cdot H$$

from (2.1.5) $\left|d1_i - d_i\right| = \left|y_i - w_i\right|$ $\quad\left|d1_i - d_i\right| \le C\cdot H^p$

Since $u1_i$ is the solution of TU1=Y and u_i is the solution of TU=W, therefore $u1_i - u_i$ is the solution of T(U1-U)=Y-W and $\left\|U1 - U\right\| \le \dfrac{\left\|Y - W\right\|}{2h}$ by Lemma 2.3.1.

Thus, we have $\left\|U1 - U\right\| \le 6\cdot\dfrac{\left\|\dfrac{y_{i+1} - w_{i+1}}{h_i} - \dfrac{y_i - w_i}{h_i} - \dfrac{y_i - w_i}{h_{i-1}} + \dfrac{y_{i-1} - w_{i-1}}{h_{i-1}}\right\|}{h_i + h_{i+1}} \le C\cdot\dfrac{H^p}{h^2}$ and

if we abuse the "C", then $\left|a1_i - a_i\right| \le C\cdot\dfrac{H^P}{h^3}$ $\quad\left|b1_i - b_i\right| \le C\cdot\dfrac{H^p}{h^2}$ $\quad\left|c1_i - c_i\right| \le C\cdot\dfrac{H^p}{h^3} + C\cdot\dfrac{H^P}{h^2}H$

Substituting into (2.3.5) we have:

$$\left|g_i(x) - s_i(x)\right| \le H^p\left[c_3\cdot\left(\frac{H}{h}\right)^3 + c_2\cdot\left(\frac{H}{h}\right)^2 + c_1\cdot\frac{H}{h} + c_0\right]$$

Substituting (2.3.4) and (2.3.5) into (2.3.3) we get (2.3.2). This prove the theorem.

Remark of Theorem

!. The theorem shows: the error of continuous approxomate solution consists of 2 parts. One is caused by interpolation and another one is caused by discrete approximate method.

2. The accuracy of continuous approximate solution can not be higher than the accuracy of the spline interpolation. In this case, If p<4, then its accurary is p; if $p \ge 4$, then its accurary is 4.

2.4 Convergence of Continuous Approximate Solution

In this part we'll discuss the convergence of the continuous approximate solution. And, the result is stated in the following theorem.

Theorem 2.4.1 *For initial value problem (1.2.1) if*

(1) f(x,y) is in C^4 *([a,b]X*$(-\infty,\infty)$*),*

(2) M_n *is a sequence of quasi-uniform simple mesh of [a,b],* M_n *:* $a = x_{0,n} < x_{1,n} < \dots\dots\dots < x_{n,n} = b$

with $\dfrac{H_n}{h_n} \le c < \infty$ *where* $h_{i,n} = x_{i+1,n} - x_{i,n}$ *i = 0,1,....,n-1* H_n*=max{* $h_{i,n}$ *}* h_n*=min {* $h_{i,n}$ *}*

(3) { $(x_{i,n}, w_{i,n})$*, i = 0,1,.....,n } is an approximate solution on* M_n *produced by a discrete approximate method with error order p*

(4) s_n*(x) is the natural cubic spline interpolant for data set in (3)*

then $s_n(x)$ *----> y(x) on [a,b] as* H_n *----> 0*

Proof: From (2.3.2) we have:

$$\left|\left|y(x) - s_n(x)\right|\right| \le K\cdot\left|\left|y^{(4)}\right|\right|\cdot\left(H_n\right)^4 + \left(H_n\right)^p\left[c_3\cdot\left(\frac{H_n}{h_n}\right)^3 + c_2\cdot\left(\frac{H_n}{h_n}\right)^2 + c_1\cdot\frac{H_n}{h_n} + c_0\right]$$

It is obvious: || y(x) - $s_n(x)$ || ----> 0 as H_n ----> 0

So, $s_n(x)$ convergences to y(x).

3. PC Approach for Constructing a Continuous Approximate Solution

In this part we'll offer three approaches to perfom the construction of the continuous approxi- mate solution in Section 2.1 which are based on PC and mathematical software "Mathcad 12 ".

3.1 The Approach Based on Program Function and Cubic Spline's Routines

This approach consists of 2 steps: first write a program RK4 which performs the Runge-Kutta method of order 4 to get discrete approximate solution and second use a built in cubic spline routine "lspline and interp" to get continuous approximate solution.

The program is as follows:

$$
\mathrm{RK4}(f,a,\alpha,b,h) := \left|
\begin{array}{l}
n \leftarrow \dfrac{b-a}{h} \\
x_0 \leftarrow a \\
w_0 \leftarrow \alpha \\
\text{for } i \in 0..n-1 \\
\quad \left|
\begin{array}{l}
x_{i+1} \leftarrow x_i + h \\
k1 \leftarrow h \cdot f(x_i, w_i) \\
k2 \leftarrow h \cdot f\left(x_i + \dfrac{h}{2}, w_i + \dfrac{k1}{2}\right) \\
k3 \leftarrow h \cdot f\left(x_i + \dfrac{h}{2}, w_i + \dfrac{k2}{2}\right) \\
k4 \leftarrow h \cdot f(x_i + h, w_i + k3) \\
w_{i+1} \leftarrow w_i + \dfrac{1}{6} \cdot (k1 + 2k2 + 2k3 + k4)
\end{array}
\right. \\
s \leftarrow \mathrm{augment}(x, w) \\
s
\end{array}
\right.
$$

In this program,
Input are function f(x,y), endpoints of interval [a,b], initial function value and step size h;
Output is a matrix of (n+1)X2 which 1st column is x-values and 2nd column is y-values.

Now, let's work out an example to illustrate this approach.

Example 3.1.1 Given: initial value problem y' $= 3 \cdot \cos(y - 3x)$ and y(0) = /2 x in I = [0,2]

Find:its continuous approximate solution on I with step size h = 0.2.

Solution: 1st Use RK4 to find its discdete approximate solution

Input: $f(x,y) := 3 \cdot \cos(y - 3 \cdot x)$ $\quad a := 0$ $\quad b := 2$ $\quad \alpha := \dfrac{\pi}{2}$ $\quad h := 0.2$

Call RK4 and define its discrete approximate solution by a matrix B: $B := \mathrm{RK4}(f, a, \alpha, b, h)$

Then, we get the discrete approximate solution :

$$B^T = \begin{pmatrix} 0 & 0.2 & 0.4 & 0.6 & 0.8 & 1 & 1.2 & 1.4 & 1.6 & 1.8 & 2 \\ 1.571 & 1.718 & 2.054 & 2.486 & 2.972 & 3.49 & 4.028 & 4.58 & 5.142 & 5.71 & 6.284 \end{pmatrix}$$

2nd, Use built-in routine "lspline" to get natural cubic spline interpolant.

$$vx := B^{\langle 0\rangle} \qquad vy := B^{\langle 1\rangle} \qquad vs := \mathrm{lspline}(vx, vy)$$

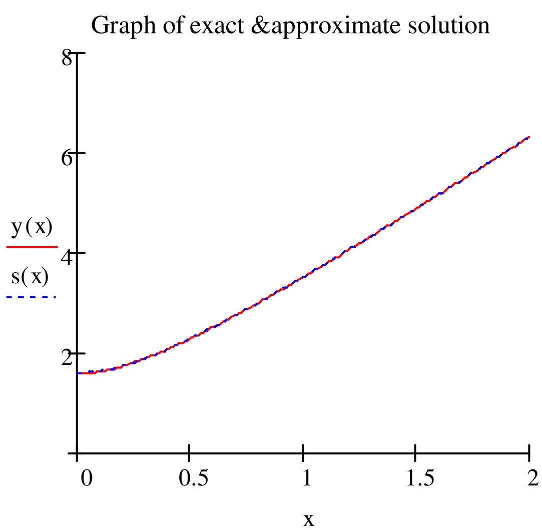
3rd, use "interp"to define the continuous approximate solution by above natural cubic spline interpolant.

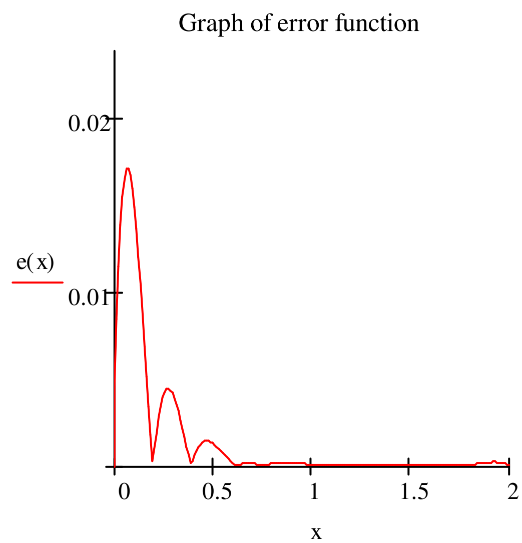
This is our desired continuous approximate solution: $s(x) := \mathrm{interp}(vs, vx, vy, x)$

In this example the IVP has a exact solution: $y(x) := 3\cdot x + 2\cdot \mathrm{acot}(3\cdot x + 1)$

Now, let's compare them by graphing both functions and their error function: $e(x) := |y(x) - s(x)|$

$x := 0, 0.01 .. 2$





We find some error: $e(0.12) = 0.014$ $\qquad e(0.5) = 1.31\times 10^{-3}$ $\qquad e(1.85) = 1.366\times 10^{-4}$

Remark:

1.In this approach the program RK4 can be replaced by built-in routine of differential equation solver "rkfixed" or "rkadpt". The approach even is simple.
2 In Mathcad there're 3 built-in routines for cubic spline interpolation: "lspline" for natural spline; "pspline" for parabolic endpoints; "cspline" for cubic endpoints(or "not a knot conditions). We can choose appropriate one according to the boundary conditions.
3.Although we have a function s(x) as the continuous approximate solution, we have no expression of the function s(x). This is a defect.

Now, we use an example to illustrate above remark 1. This example is same as Example 3.1.1 but we use the routine "rkfixed" to find discrete approximate solution rather than program Rk4.

Example 3.1.2 Given: initial value problem y' $= 3\cdot\cos(y - 3x)$ and y(0) = □/2 x in I = [0,2]
Find:its continuous approximate solution on I with step size h = 0.2.
Solution: 1st use "rkfixed" to find discrete approximate solution of given IVP.

$$y_0 := \frac{\pi}{2} \qquad D(x,y) := 3\cdot\cos\left(y_0 - 3\cdot x\right)$$

Call the routine and the discrete approximate solution is indicated in matrix W: $W := \text{rkfixed}(y, 0, 2, 10, D)$

then, we get same result in B:

$$W^T = \begin{pmatrix} 0 & 0.2 & 0.4 & 0.6 & 0.8 & 1 & 1.2 & 1.4 & 1.6 & 1.8 & 2 \\ 1.571 & 1.718 & 2.054 & 2.486 & 2.972 & 3.49 & 4.028 & 4.58 & 5.142 & 5.71 & 6.284 \end{pmatrix}$$

2nd, We use this discrete approximate solution and "lspline" to find the natural cubic spline interpolant.

$$vx := W^{\langle 0\rangle} \qquad vy := W^{\langle 1\rangle} \qquad vs := \text{lspline}(vx, vy)$$

3rd, use this result and "interp" to get our desired continuous approximate solution.

$$s1(x) := \text{interp}(vs, vx, vy, x)$$

Remark:
Writing the program for discrete approximate solution still is necessary although there're some routines available since not every discrete approximate method's routine is available. Moreover, sometime we need the discrete approximate method with order p greater than 4 to guarantee our desired accuracy.

3.2 The Approach Based on Program Finction and Routine of "Solve Block"

This approach consists of following steps: 1. Write a program to get discrete approximate solution; 2. Define the natural cubic spline interpolant by (1.3.1) and get the system of linear equations in 4n unknown coefficients by (2.1.0); 3. Use built-tn routine "solve block" to solve above system; 4. Use the result to construct the natural cubic spline interpolant -- the desired continuous approximate solution.

Now, let's work out an example to illustrate this approach.

Example 3.2 Given: $y' = 1 + (x-y)^2$ and $y(2) = 1$ x in I = [2,3]

Find: its continuous approximate solution on I with step size h = 0.25

Solution: 1st, get its discrete approximate solution by RK4 in section 3.1

$$\mathrm{RK4}(f, a, \alpha, b, h) := \left| \begin{array}{l} n \leftarrow \dfrac{b-a}{h} \\ x_0 \leftarrow a \\ w_0 \leftarrow \alpha \\ \text{for } i \in 0..n-1 \\ \quad \left| \begin{array}{l} x_{i+1} \leftarrow x_i + h \\ k1 \leftarrow h \cdot f(x_i, w_i) \\ k2 \leftarrow h \cdot f\left(x_i + \dfrac{h}{2}, w_i + \dfrac{k1}{2}\right) \\ k3 \leftarrow h \cdot f\left(x_i + \dfrac{h}{2}, w_i + \dfrac{k2}{2}\right) \\ k4 \leftarrow h \cdot f(x_i + h, w_i + k3) \\ w_{i+1} \leftarrow w_i + \dfrac{1}{6} \cdot (k1 + 2k2 + 2k3 + k4) \end{array} \right. \\ s \leftarrow \mathrm{augment}(x, w) \\ s \end{array} \right.$$

In this program,
Input are function f(x,y), endpoints of interval [a,b], initial function value □ and step size h;
Output is a matrix of (n+1)X2 which 1st column is x-values and 2nd column is y-values.

Input: $f(x,y) := 1 + (x-y)^2$ $\quad a := 2$ $\quad \alpha := 1$ $\quad b := 3$ $\quad h := 0.25$

Call RK4 and define its discrete approximate solution by B: $B := \mathrm{RK4}(f, a, \alpha, b, h)$

Then, the discrete approximate solution is:

$$B^T = \begin{pmatrix} 2 & 2.25 & 2.5 & 2.75 & 3 \\ 1 & 1.45 & 1.833 & 2.179 & 2.5 \end{pmatrix}$$

$$x_0 := \left(B^{\langle 0 \rangle}\right)_0 \quad x_1 := \left(B^{\langle 0 \rangle}\right)_1 \quad x_2 := \left(B^{\langle 0 \rangle}\right)_2 \quad x_3 := \left(B^{\langle 0 \rangle}\right)_3 \quad x_4 := \left(B^{\langle 0 \rangle}\right)_4$$

$$w_0 := \left(B^{\langle 1 \rangle}\right)_0 \quad w_1 := \left(B^{\langle 1 \rangle}\right)_1 \quad w_2 := \left(B^{\langle 1 \rangle}\right)_2 \quad w_3 := \left(B^{\langle 1 \rangle}\right)_3 \quad w_4 := \left(B^{\langle 1 \rangle}\right)_4$$

$h_0 := x_1 - x_0$ $\qquad h_1 := x_2 - x_1$ $\qquad h_2 := x_3 - x_2$ $\qquad h_3 := x_4 - x_3$

2nd, Using above data we define the natural cubic spline interpolant s(x) which is equal to

$s0(x) = a_0(x-x_0)^3 + b_0(x-x_0)^2 + c_0(x-x_0) + d_0 \qquad x \text{ in } [x_0, x_1]$

$s1(x) = a_1(x-x_1)^3 + b_1(x-x_1)^2 + c_1(x-x_1) + d_1 \qquad x \text{ in } [x_1, x_2]$

$s2(x) = a_2(x-x_2)^3 + b_2(x-x_2)^2 + c_2(x-x_2) + d_2 \qquad x \text{ in } [x_2, x_3]$

$s3(x) = a_3(x-x_3)^3 + b_3(x-x_3)^2 + c_3\cdot(x-x_3) + d_3 \qquad x \text{ in } [x_3, x_4]$

Now, the problem is reduced to finding these coefficients from the system which come from (1.3.2.a)----(1.3.2.e).

3rd, Use "solve block" to find these coefficients which structure is "Guess -- Given -- Find".

Guess

$a_0 := 0 \quad b_0 := 0 \quad c_0 := 0 \quad d_0 := 0 \quad a_1 := 0 \quad b_1 := 0 \quad c_1 := 0 \quad d_1 := 0$

$a_2 := 0 \quad b_2 := 0 \quad c_2 := 0 \quad d_2 := 0 \quad a_3 := 0 \quad b_3 := 0 \quad c_3 := 0 \quad d_3 := 0$

Given

$d_0 = w_0 \qquad d_1 = w_1 \qquad d_2 = w_2 \qquad d_3 = w_3 \qquad a_3(h_3)^3 + b_3(h_3)^2 + c_3 h_3 + d_3 = w_4$

$a_0(h_0)^3 + b_0(h_0)^2 + c_0 h_0 + d_0 = d_1 \qquad a_1(h_1)^3 + b_1(h_1)^2 + c_1 h_1 + d_1 = d_2$

$a_2(h_2)^3 + b_2(h_2)^2 + c_2 h_2 + d_2 = d_3$

$3a_0(h_0)^2 + 2\cdot b_0 h_0 + c_0 = c_1 \qquad 3a_1(h_1)^2 + 2\cdot b_1 h_1 + c_1 = c_2 \qquad 3a_2(h_2)^2 + 2\cdot b_2 h_2 + c_2 = c_3$

$6a_0 h_0 + 2\cdot b_0 = 2\cdot b_1 \qquad 6a_1 h_1 + 2\cdot b_1 = 2\cdot b_2 \qquad 6a_2 h_2 + 2\cdot b_2 = 2\cdot b_3$

$2\cdot b_0 = 0 \qquad 6a_3 h_3 + 2\cdot b_3 = 0$

$C := \text{Find}(a_0, b_0, c_0, d_0, a_1, b_1, c_1, d_1, a_2, b_2, c_2, d_2, a_3, b_3, c_3, d_3)$

$C^T = \blacksquare$

$a_0 := C_0 \quad b_0 := C_1 \quad c_0 := C_2 \quad d_0 := C_3 \quad a_1 := C_4 \quad b_1 := C_5 \quad c_1 := C_6 \quad d_1 := C_7$

$a_2 := C_8 \quad b_2 := C_9 \quad c_2 := C_{10} \quad d_2 := C_{11} \quad a_3 := C_{12} \quad b_3 := C_{13} \quad c_3 := C_{14} \quad d_3 := C_{15}$

4th, Use these coefficient to construct the natural cubic spline interpolant.

$$s0(x) := -0.996(x-2)^3 + 0(x-2)^2 + 1.862(x-2) + 1 \qquad x \text{ in } [2,2.25]$$

$$s1(x) := 0.731(x-2.25)^3 - 0.747(x-2.25)^2 + 1.675(x-2.25) + 1.45 \qquad x \text{ in } [2.25,2.5]$$

$$s2(x) := -0.027(x-2.5)^3 - 0.212(x-2.5)^2 + 1.436(x-2.5) + 1.833 \qquad x \text{ in } [2.5,2.75]$$

$$s3(x) := 0.31(x-2.75)^3 - 0.233(x-2.75)^2 + 1.324(x-2.75) + 2.179 \qquad x \text{ in } [2.75,3]$$

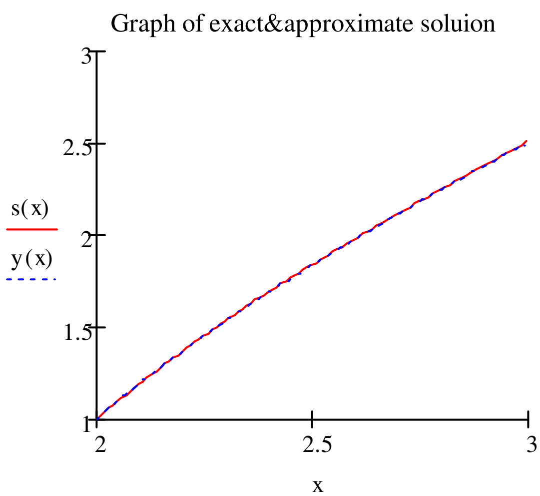
Or we can combine them into one function by built-in routine "if". We define:

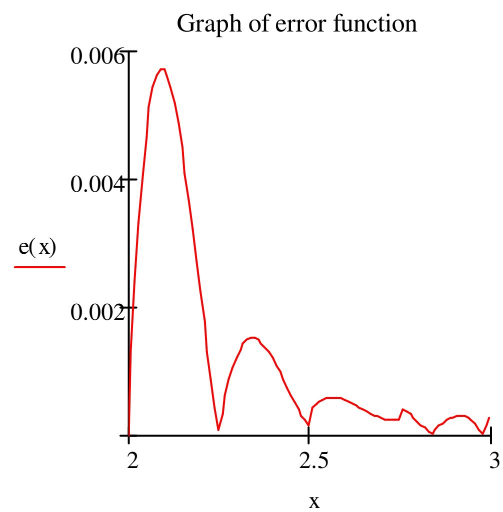
$$s(x) := \text{if}(2 \le x \le 2.25, s0(x), \text{if}(2.25 \le x \le 2.5, s1(x), \text{if}(2.5 \le x \le 2.75, s2(x), s3(x))))$$

Also, this initial value problem has unique solution: $y(x) := x - \dfrac{1}{x-1}$

Now, we compare them by graphing them and their error function: $e(x) := |y(x) - s(x)|$

$x := 2, 2.01 .. 3$





Remark:
1. This approach can get an expression of the continuous approximate solution but need to do more work ourself .
2. This approach is limited by the capacity of software. For example Mathcad can solve the system of equations up to 60 unknowns so it can find a cubic spline of at most 15 pieces.
3. We can use another routine "solve" of symbolic operation instead of "solve block" in this approach.

3.3 The Approach Based on Two Programs
This approach consists of 3 steps: 1. Use a program to get discrete approximate solution of initial value problem; 2. Use second program and above data to get coefficients of the natural cubic spline interpolant; 3. Use the result to construct the continuous approximate solution.

Now, let's work out an example to illustrate this approach.

Example 3.3 Given: $y' = \frac{y}{x} - \left(\frac{y}{x}\right)^2$ and $y(1) = 1$ x in $I = [\,1,3\,]$

Find: its continuous approximate solution on I with step size h = 0.2

Solution: 1st, we use RK4 in previous section to get its discrete approximate solution.

Input: $f(x,y) := \frac{y}{x} - \left(\frac{y}{x}\right)^2$ $\quad a := 1 \quad \alpha := 1 \quad b := 3 \quad h := 0.2$

$$
\mathrm{RK4}(f,a,\alpha,b,h) := \left|\begin{array}{l}
n \leftarrow \dfrac{b-a}{h} \\
x_0 \leftarrow a \\
w_0 \leftarrow \alpha \\
\text{for } i \in 0..n-1 \\
\quad \left|\begin{array}{l}
x_{i+1} \leftarrow x_i + h \\
k1 \leftarrow h \cdot f(x_i, w_i) \\
k2 \leftarrow h \cdot f\left(x_i + \dfrac{h}{2}, w_i + \dfrac{k1}{2}\right) \\
k3 \leftarrow h \cdot f\left(x_i + \dfrac{h}{2}, w_i + \dfrac{k2}{2}\right) \\
k4 \leftarrow h \cdot f(x_i + h, w_i + k3) \\
w_{i+1} \leftarrow w_i + \dfrac{1}{6} \cdot (k1 + 2k2 + 2k3 + k4)
\end{array}\right. \\
w
\end{array}\right.
$$

In this program,
Input are function f(x,y), endpoints of interval [a,b], initial function value □ and step size h;
Output is a matrix of (n+1)X1 which is approximate solution values of y(x).

Then, we have the discrete approximate solution in B: $B := \mathrm{RK4}(f,a,\alpha,b,h)$

$$B^T = (1 \;\; 1.015 \;\; 1.048 \;\; 1.088 \;\; 1.134 \;\; 1.181 \;\; 1.23 \;\; 1.28 \;\; 1.33 \;\; 1.38 \;\; 1.43)$$

2nd, in this example the step sizes are same h = 0.2 , so the interval is divided into 10 subintervals i.e. the natural cubic spline interpolant consists of 10 piece of cubic polynomial. And, we need to find 40 coefficients for constructing it. We use the program Cf which solve the system of (2.1.1) -- (2.1.5), to get the coefficients of the natural cubic spline interpolant. This program applies LU factorization to solve (2.1.1) to get u_i and then find { a_i, b_i, c_i, d_i } by u_i from (2.1.2) -- (2.1.5).

The Cf program is as follows:

$$
Cf(f,a,\alpha,b,h) := \left|
\begin{array}{l}
n \leftarrow \dfrac{b-a}{h} \\
\text{for } i \in 0..n \\
\quad w_i \leftarrow B_i \\
\text{for } i \in 1..n-1 \\
\quad v_i \leftarrow \dfrac{6}{h}\cdot\left(w_{i+1} - 2\cdot w_i + w_{i-1}\right) \\
l_0 \leftarrow 1 \\
\mu_0 \leftarrow 0 \\
z_0 \leftarrow 0 \\
\text{for } i \in 1..n-1 \\
\quad \left|
\begin{array}{l}
l_i \leftarrow 2\cdot h - h\cdot\mu_{i-1} \\
\mu_i \leftarrow \dfrac{h}{l_i} \\
z_i \leftarrow \dfrac{v_i - h\cdot z_{i-1}}{l_i}
\end{array}\right. \\
l_n \leftarrow 1 \\
z_n \leftarrow 0 \\
u_n \leftarrow 0 \\
\text{for } j \in (n-1)..0 \\
\quad u_j \leftarrow z_j - \mu_j\cdot u_{j+1} \\
\text{for } k \in 0..(n-1) \\
\quad \left|
\begin{array}{l}
a_k \leftarrow \dfrac{u_{k+1} - u_k}{6\cdot h} \\
b_k \leftarrow \dfrac{u_k}{2} \\
c_k \leftarrow \dfrac{w_{k+1} - w_k}{h} - \dfrac{u_{k+1} + 2\cdot u_k}{6}\cdot h \\
d_k \leftarrow w_k
\end{array}\right. \\
s \leftarrow \text{augment}(a,b,c,d) \\
s
\end{array}\right.
$$

We get these coefficients in the matrix which is indicated by E: $E := Cf(f, a, \alpha, b, h)$

$$E^T = \begin{pmatrix} 1.409 & -2.023 & 1.473 & -1.422 & 1.126 & -0.962 & 0.721 & -0.527 & 0.304 & -0.099 \\ 0 & 0.846 & -0.369 & 0.515 & -0.338 & 0.337 & -0.24 & 0.193 & -0.123 & 0.06 \\ 0.018 & 0.075 & 0.219 & 0.18 & 0.26 & 0.215 & 0.267 & 0.232 & 0.262 & 0.242 \\ 1 & 1.015 & 1.048 & 1.088 & 1.134 & 1.181 & 1.23 & 1.28 & 1.33 & 1.38 \end{pmatrix}$$

3rd, We use above result to construct the continuous approximate solution s(x) which is equal to:

$s0(x) := 1.409(x-1)^3 + 0\cdot(x-1)^2 + 0.018(x-1) + 1$ x in [1,1.2]

$s1(x) := -2.203(x-1.2)^3 + 0.846(x-1.2)^2 + 0.075(x-1.2) + 1.015$ x in [1.2,1.4]

$s2(x) := 1.473(x-1.4)^3 - 0.369(x-1.4)^3 + 0.219(x-1.4) + 1.048$ x in [1.4,1.6]

$s3(x) := -1.422(x-1.6)^3 + 0.515(x-1.6)^2 + 0.18(x-1.6) + 1.088$ x In [1.6,1.8]

$s4(x) := 1.126(x-1.8)^3 - 0.338(x-1.8)^2 + 0.26(x-1.8) + 1.134$ x in [1.8,2]

$s5(x) := -0.962(x-2)^3 + 0.337(x-2)^2 + 0.215(x-2) + 1.181$ x in [2,2.2]

$s6(x) := 0.721(x-2.2)^3 - 0.24(x-2.2)^2 + 0.267(x-2.2) + 1.23$ x in [2.2,2.4]

$s7(x) := -0.527(x-2.4)^3 + 0.193(x-2.4)^2 + 0.232(x-2.4) + 1.28$ x in [2.4,2.6]

$s8(x) := 0.304(x-2.6)^3 - 0.123(x-2.6)^2 + 0.262(x-2.6) + 1.33$ x in [2.6,2.8]

$s9(x) := -0.099(x-2.8)^3 + 0.06(x-2.8)^2 + 0.242(x-2.8) + 1.38$ x in [2.8,3]

Or, we can combine by built-in routine "if".

$u1(x) := if(1 \le x \le 1.2, s0(x), if(1.2 \le x \le 1.4, s1(x), if(1.4 \le x \le 1.6, s2(x), s3(x))))$

$u2(x) := if(1.8 \le x \le 2, s4(x), if(2 \le x \le 2.2, s5(x), if(2.2 \le x \le 2.4, s6(x), s7(x))))$

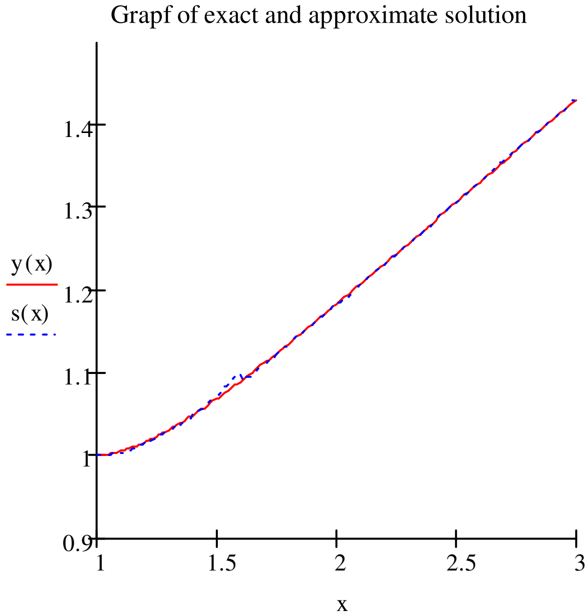
$u3(x) := if(2.6 \le x \le 2.8, s8(x), s9(x))$

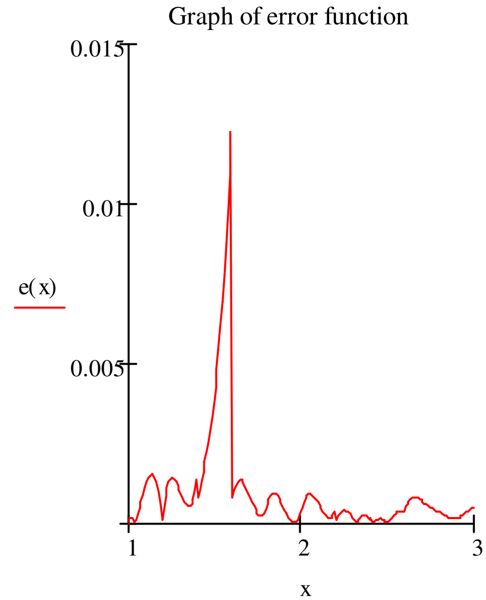
$s(x) := if(1 \le x \le 1.8, u1(x), if(1.8 \le x \le 2.6, u2(x), u3(x)))$

This is our desired continuous approximate solution.
And, this initial value problem has an exact solution y(x).

Now, let compare the continuous approximate solution and exact solution by graphing them and their error function e(x).

$x := 1, 1.01 .. 3$ $y(x) := \dfrac{x}{1 + \ln(x)}$ $e(x) := |y(x) - s(x)|$





Remark:

1. This approach can find the expression of the continuous approximate solution.
2. This approach is based on two programs. In 2nd program we use LU factorization to solve (2.1.1), of cause we can use other ways and other program.

4. REFERENCE

[1] Birhoff, Garrett & De Boor, Carl: "Error Bounds for Spline Interpolation", Journal of Mathematics and Mechanics, Vol 13, No 5, 1964.

[2] Burden, Richard L & Faires, J. Douglas: "Numerical Analysis" 8th edition, Brooks/Cole, Tomson Learning Inc. 2004.

[3] Davis,P.J.: "Interpolation and approximation", Blaisdell, New York, 1963.

[4] De Boor, Carl:"A practical guide to spline", Spring Verlag Inc., New York, 2001.

[5] E.Hairer, S.P.Norsett, G.W.Wanner: "Solving Ordinary Differential Equstion I" (nonstiff problem) Springer, 1993 Vol. 1

[6] E.Hairer, S.P.Norsett, G.W.Wanner: "Solving Ordinary Differential Equstion II" (stiff and differential algebraic problem) Springer, 1996.

[7] Loscalzo, F.R. 7 Tallot,T.D.:"Spline function approximation for solutions of ordinary differential equations", SIAM J., Numerical Analysis 4, 1967, p433 - 445.

[8] Schoenberg, I. J: " On spline functions", In Tech Summary Rep. 625. Mathematics Research Center, University of Wisconsin, Madison, 1966.

[9] Wu, D.T.:"Teaching Numerical Analysis with Mathcad", The International Journal of Computer Algebra in Mathematical Edication, 10 No 1, 2003.

[10] WU, D.T.: "Mathcad's Program Function and Application in teaching of Mathmatics" in Electronic Proceeding of ICMCM 16, Aug. 2004