New simulation of interpolation points using polynomials system to apply numerical approximation

Mustafa Abbas Fadhel

Department of Sciences, College of Basic Education, University of Al-Muthanna, Samawa, Iraq

mustafa@mu.edu.iq

Received: 15/12/2015 ,Accepted: 12/3/2016 , Bublished:20/9/2016

Abstract In this paper, we have taken some interpolation points and found a polynomial passing through it by simple way. Using a slave of a system of equations composed of interpolation points, such that the y-axis of this point will be as solution of equations, and x-axis will be equations that find it by composition interpolation points with constant points to make a system of polynomials, by solving this system we get a result of the constant points that will be shaped interpolation polynomial.

Keywords: Interpolation, slave of a system, numerical approximation.

Introduction

Many times we find many mathematical problems need solutions without sufficient data, many scientists works to finding logical solutions, but these solutions always be accompanied by a different error rates. In this paper we will examine a new method of Interpolation by another way to defining curves by using new interpolation polynomial. We will see that this is far more useful than either the Cartesian description or the polar form. Although we shall only study planar curves the polynomial description can be generalized to the description of spatial curve to find interpolation points.

Basic Definitions

In this section, the basic definitions with properties are presented, we will show the meaning of interpolation and represents it by spline method interpolation, and show that some of characteristics to this method.

The concept of meaning interpolation, linear Spline interpolating and cubic spline were introduced by Levy [1], Suli, Mayers [2] and Stoer, Bulirsch [3], respectively.

Definition 2.1 An unknown function $f(x)$ for which someone supplies you with its (exact) values at $(n+1)$ distinct points $x_0 < x_1 < \cdots < x_n$, $i.e., f(x_0), \dots, f(x_n)$ are given. The interpolation problem is to construct a function $Q(x)$ that passes through these points, i.e., to find a function $Q(x)$ such that the interpolation requirements $Q(x_j) = f(x_j), 0 \le j \le n$, are satisfied.

Definition 2.2: Suppose that f is a real-valued function, defined and continuous on the closed interval $[a,b]$. Further, $let\ K = \{x_0, \dots, x_m\}$ be a subset of $[a,b]$, with $a = x_0 < x_1 < \cdots < x_m = b$, $m \ge 2$. The linear spline sL, interpolating f at the points x_i, is defined by

$$sL(x) = \frac{x_i - x}{x_i - x_{i-1}} f(x_{i-1}) + \frac{x - x_{i-1}}{x_i - x_{i-1}} f(x_i),\ x \in [x_{i-1}, x_i],\ i = 1, 2, \dots, m. \quad (1)$$

Definition 2.3 A cubic spline (function) $\mathbf{S}_\Delta$ on Δ is a real function $\mathbf{S}_\Delta: [\mathbf{a}, \mathbf{b}] \to \mathbb{R}$ with the properties:

$S_\Delta \in C^2[a,b]$, that is, S_Δ is twice continuously differentiable on $[a,b]$.
S_Δ coincides on every subinterval $[x_i, x_{i+1}], i = 0,1,\dots,n-1$, with a polynomial of degree three.
Thus a cubic spline consists of cubic polynomials pieced together in such a fashion that their values and those of their first two derivatives coincide at the knots x_i, $i = 1,\dots,n-1$.
Proposed Method (MAF Interpolation)
When we have $n+1$ interpolation points [let as :$W_0, W_1, \dots, W_n$; $W_n = (x_n, y_n)$], then to find a interpolation polynomial that passes through these points, then we have $n+1$ values of the polynomial, we can choose this points as follow:
Let $P(x) = a_0x^n + a_1x^{n-1} + \cdots + a_{n-1}x + a_n$

Then the solution of system will be as

To find the value of $a_0 \cdots a_n$, suppose that:

$$\begin{aligned} y_0 &= a_n x_0{}^n + \cdots + a_1 x_0 + a_0 \\ y_1 &= a_n x_1{}^n + \cdots + a_1 x_1 + a_0 \\ &\vdots \qquad \vdots \qquad \ddots \qquad \vdots \qquad \vdots \\ y_n &= a_n x_n{}^n + \cdots + a_1 x_n + a_0 \end{aligned} \qquad (2)$$

We can solve this system by suppose:

$$A = \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{bmatrix}, T = \begin{bmatrix} x_0{}^n & x_0{}^{n-1} & \cdots & x_0{}^0 \\ x_1{}^n & x_1{}^{n-1} & \cdots & x_1{}^0 \\ \vdots & \vdots & \ddots & \vdots \\ x_n{}^n & x_n{}^{n-1} & \cdots & x_n{}^0 \end{bmatrix}, S = \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{bmatrix} \qquad (3)$$

$$A = T^{-1}S \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} x_0{}^n & x_0{}^{n-1} & \cdots & x_0{}^0 \\ x_1{}^n & x_1{}^{n-1} & \cdots & x_1{}^0 \\ \vdots & \vdots & \ddots & \vdots \\ x_n{}^n & x_n{}^{n-1} & \cdots & x_n{}^0 \end{bmatrix}^{-1} \times \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{bmatrix} \qquad (4)$$

From equation (4), we can find the value of $a_0, \dots, a_n$, therefor the interpolation polynomial will be as
$P(x) = a_0x^n + \cdots + a_{n-1}x + a_n$.

Comparing the Results by Test Example

In this section, we will define a method which is less error by finding a ratio error for two test examples by cubic spline and MAF Interpolation than compare a results for it.
Notes [4]:
We will find the ratio error in the test examples by using $L_2 metric$ definition as the following formula:

$$e_n = f - p_n \,, than \quad \|e_n\|_2 = \|f - p_n\|_2 = \left(\int |f(x) - p_n(x)|^2 dx\right)^{\frac{1}{2}} \qquad (5)$$

Examples: Table 4.1 shows many data for many functions, as follow:

Table 4.1 : Interpolation points for many function

$f(x) = x$	x^2	$x^6 - 1$	$(x+3)^5$	e^{2x}	$sinh\frac{5}{2}x$	$k - Monotone$
-2.2114	4.8903	115.9510	0.3050	0.0120	−125.8810	−47.5604
-1.3326	1.7758	4.6001	12.8884	0.0696	−13.9723	−18.2404
0.2668	0.0712	−0.9996	372.0598	1.7051	0.7176	−0.5701
1.0023	1.0046	0.0139	$1.0269 \times 10^{+3}$	7.4231	6.0856	0

1.8893	3.5695	44.4784	$2.7940 \times 10^{+3}$	43.7547	56.2623	1

Recall that, a $K-monotone$ was given in [5], as follows:

$$f(x) = \begin{cases} (t_n - t_{n-1})^{1-n}(x - t_{n-1})_+^{n-1} & if \quad x \in [t_i, t_{i+1}]; \\ 0 & if \quad x \notin [t_i, t_{i+1}], \end{cases}$$
(6)

where $0 \le i \le n-1$, $(x - t_{n-1})_+^{n-1} = max\{(x - t_{n-1})^{n-1}, 0\}$.

```
t0= -2.2114 ; t1 = -1.3326  ; t2 = 0.2668 ; t3 =
1.0023 ; t4 = 1.8893 ;
t= [ t0  t1  t2  t3  t4 ];
y= t.^2 ;
y0= y(1,1) ; y1= y(1,2) ; y2= y(1,3) ; y3= y(1,4) ;
y4= y(1,5);
h0= t1-t0 ; h1=t2-t1 ; h2=t3-t2 ; h3=t4-t3 ;
r0= (y1-y0)/h0;
r1= (y2-y1)/h1;
r2= (y3-y2)/h2;
r3= (y4-y3)/h3;
u1= 2*(h0+h1);
u2= 2*(h1+h2)-((h1.^2)/u1);
u3= 2*(h2+h3)-((h2.^2)/u2);
v1= 6*(r1-r0);
v2 = (6*(r2-r1))-((h1*v1)/u1);
v3 = (6*(r3-r2))-((h2*v2)/u2);
z4 = 0 ;
z3= (v3-(h3*z4))/u3;
z2= (v2-(h2*z3))/u2;
z1= (v1-(h1*z2))/u1;
z0= 0 ;
a0 = y0 ;
a1 = y1 ;
a2 = y2 ;
a3 = y3 ;
b0= -((h0/6)*z1)-((h0/3)*z0)+((y1-y0)/h0) ;
b1= -((h1/6)*z2)-((h1/3)*z1)+((y2-y1)/h1) ;
b2= -((h2/6)*z3)-((h2/3)*z2)+((y3-y2)/h2) ;
```

To construct approximation polynomial to above data by two methods (cubic spline interpolation and MAF interpolation) and comparing between it, we can use the following program:

Solution MATLAB Program to $\boldsymbol{x^2}$.

```
b3= -((h3/6)*z4)-((h3/3)*z3)+((y4-y3)/h3) ;
c0= z0/2 ;
c1= z1/2 ;
c2= z2/2 ;
c3= z3/2 ;
d0= (z1-z0)/(6*h0) ;
d1= (z2-z1)/(6*h1) ;
d2= (z3-z2)/(6*h2) ;
d3= (z4-z3)/(6*h3);
x=t0:.001:t4;
e0= t0:.001:t1;
e1= t1:.001:t2;
e2= t2:.001:t3;
e3= t3:.001:t4;
Sx0=(d0*((e0-t0).^3))+(c0*((e0-t0).^2))+(b0*(e0-
t0))+a0;
Sx1=(d1*((e1-t1).^3))+(c1*((e1-t1).^2))+(b1*(e1-
t1))+a1;
Sx2=(d2*((e2-t2).^3))+(c2*((e2-t2).^2))+(b2*(e2-
t2))+a2;
Sx3=(d3*((e3-t3).^3))+(c3*((e3-t3).^2))+(b3*(e3-
t3))+a3;
fx= x.^2 ;
T= [(t0)^4 (t0)^3 (t0)^2 t0 1 ; (t1)^4 (t1)^3 (t1)^2 t1
1 ; (t2)^4 (t2)^3 (t2)^2 t2 1 ; (t3)^4 (t3)^3 (t3)^2 t3
1; (t4)^4 (t4)^3 (t4)^2 t4 1];
Sx=[t0 ; t1 ; t2 ; t3 ; t4];
Sy=[y]' ;
b=[ inv(T)*Sy ];
```

```
px =  (b(1,1)*x.^4)+ (b(2,1)*x.^3) + (b(3,1)*x.^2) +
(b(4,1)*x) + b(5,1);
plot(t,y,'o',x,fx,x,px,e0,Sx0,e1,Sx1,e2,Sx2,e3,Sx3)
syms x
fx= x.^2 ;
Sx0=(d0*((x-t0).^3))+(c0*((x-t0).^2))+(b0*(x-
t0))+a0;
Sx1=(d1*((x-t1).^3))+(c1*((x-t1).^2))+(b1*(x-
t1))+a1;
Sx2=(d2*((x-t2).^3))+(c2*((x-t2).^2))+(b2*(x-
t2))+a2;
Sx3=(d3*((x-t3).^3))+(c3*((x-t3).^2))+(b3*(x-
t3))+a3;
px =  (b(1,1)*x.^4)+ (b(2,1)*x.^3) + (b(3,1)*x.^2) +
(b(4,1)*x) + b(5,1);
ErrorS0 = sqrt(abs(int((fx-Sx0).^2,x,t0,t1)));
ErrorS1 = sqrt(abs(int((fx-Sx1).^2,x,t1,t2)));
ErrorS2 = sqrt(abs(int((fx-Sx2).^2,x,t2,t3)));
ErrorS3 = sqrt(abs(int((fx-Sx3).^2,x,t3,t4)));
ErrorSpline = ErrorS0+ErrorS1+ErrorS2+ErrorS3
EMAF = sqrt(abs(int((fx-px).^2,x,t0,t4)))
S0=collect (Sx0);
S1=collect (Sx1);
S2=collect (Sx2);
S3=collect (Sx3);
 pretty (S0)
 pretty (S1)
 pretty (S2)
 pretty (S3)
 pretty (px)
```

Not: We can find another solution to the functions by changing the program above in some parts which it write by red color.

Table 4.2 shows the Approximation by MAF polynomial for each functions:

Table 4.2: Approximation by MAF polynomial

Original function	**Approximation by MAF polynomial**
x^2	$p(x) = \frac{x^4}{36028797018963968} - \frac{x^3}{18014398509481984} + x^2 - \frac{x}{9007199254740992} + \frac{1}{36028797018963968}$
$x^6 - 1$	$p(x) = \frac{6451430166401275}{1125899906842624}x^4 - \frac{1004647450519309}{562949953421312}x^3 - \frac{218322442679429}{35184372088832}x^2 + \frac{1077173102245561}{281474976710656}x - \frac{27691884999149}{17592186044416}$
$(x+3)^5$	$p(x) = \frac{9134}{625}x^4 + \frac{105092787790493}{1099511627776}x^3 + \frac{2378178119115201}{8796093022208}x^2 + \frac{3509083709327083}{8796093022208}x + \frac{2150546687066063}{8796093022208}$

e^{2x}	$p(x) = \frac{2565023113124043}{2251799813685248}x^4 + \frac{1897690154285335}{562949953421312}x^3 + \frac{208197122546635}{140737488355328}x^2 - \frac{209546820822053}{1125899906842624}x + \frac{3556991738592701}{2251799813685248}$
$sinh\frac{5}{2}x$	$p(x) = -\frac{42118577007671}{35184372088832}x^4 + \frac{834683051305063}{70368744177664}x^3 + \frac{3725862674235969}{2251799813685248}x^2 - \frac{5122589458631795}{562949953421312}x + \frac{1580958352378945}{562949953421312}$
k − Monotone	$p(x) = \frac{7}{36028797018963968}x^4 + \frac{3226702453966523}{2251799813685248}x^3 - \frac{1212796451103993}{281474976710656}x^2 + \frac{1215585882941533}{281474976710656}x - \frac{6498035895852259}{4503599627370496}$

Table 4.2: Approximation by MAF polynomial

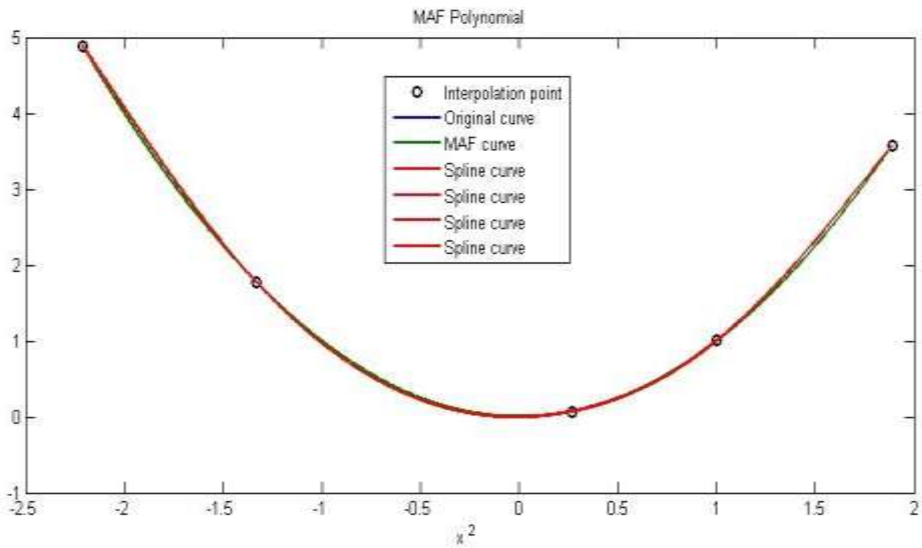
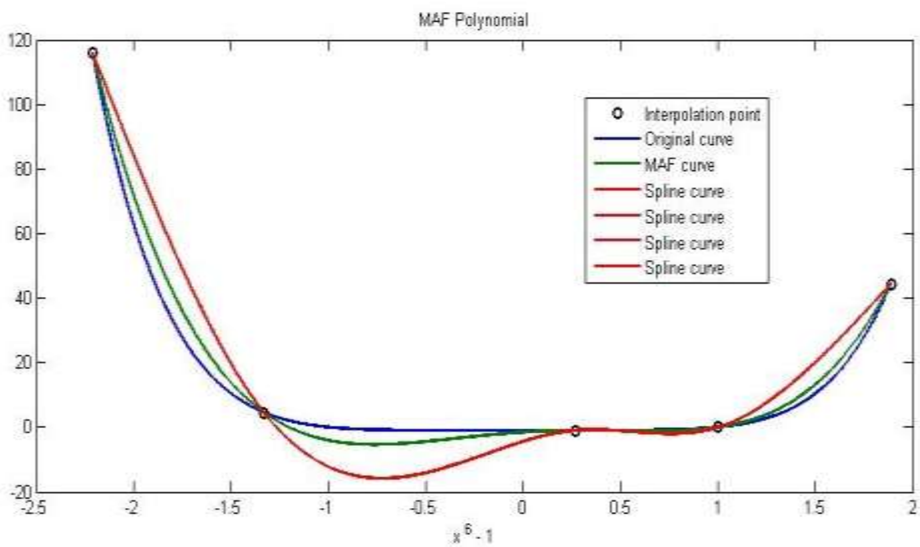
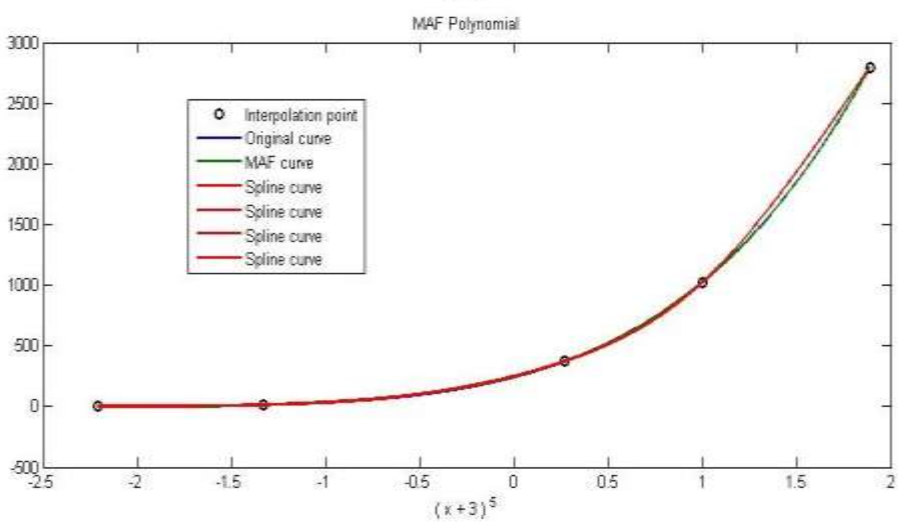
Table 4.2 shows the ratio error of cubic spline interpolation and MAF interpolation for each functions studied.

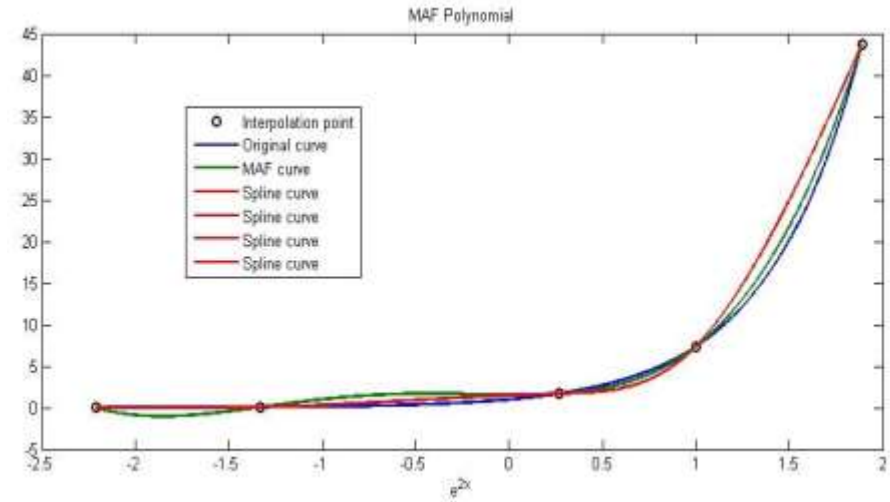
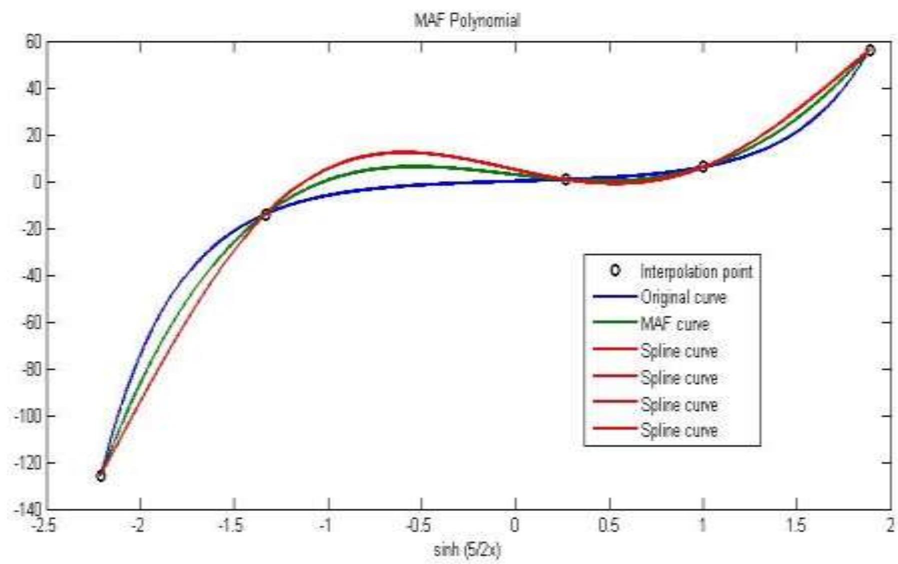
Table 4.3: ratio errors

Errors	x^2	x^6-1	$(x+3)^5$	e^{2x}	$sinh\frac{5}{2}x$	$k-Monotone$
Spline	0.1400	35.5048	72.3370	4.5002	35.6213	1.6007
MAF	9.0486×10^{-16}	7.5441	4.9106	1.8243	11.9968	4.0851×10^{-15}

Fig 4.1 shows the plot comparing of original curve, MAF interpolation and spline interpolation for each functions:

Original function	Plot comparing

Original function	Plot comparing
x^2	
$x^6 - 1$	
$(x + 3)^5$	

<table>
<tr><th>Original function</th><th>Plot comparing</th></tr>
<tr><td>e^{2x}</td><td>
</td></tr>
<tr><td>$sinh\frac{5}{2}x$</td><td>
</td></tr>
</table>

Original function	Plot comparing
$k - Monotone$	(see figure below)

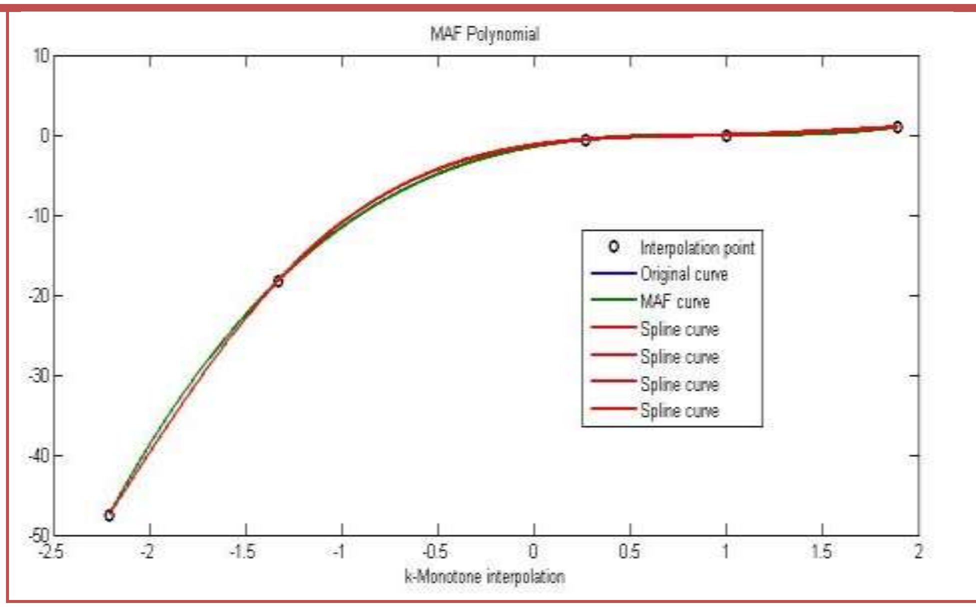


Fig. 4.1 plot comparing to each functions

Conclusion

Getting new polynomial interpolation by analytical arranged to interpolation points, so the MAF interpolation passed through these points. Through investigating with previous studies in this field, it became clearer that the MAF interpolation is more efficient by test and compare it with another best method and find the ratio error for it.

Acknowledgements

The author is grateful for hospitality at the University of Kufa. He thanks his fellows for the fruitful discussions while preparing this paper. He was partially supported by University of Al-Muthanna.

References

[1] D. Levy, Introduction to numerical analysis, University of Maryland, 2010.

[2]E. Suli and D. F. Mayers, An introduction to numerical analysis, Cambridge university press, Cambridge, UK, 2003.

[3] J. Stoer and R. Bulirsch, Introduction to numerical analysis, Translated by R. Bartels, W. Gautschi and C. Witzgall, Second edition, Springer-Verlag New York, Inc., 1993.

[4] C. J. Zarowski, An introduction to numerical analysis for electrical and computer engineers, John Wiley & Sons, Inc., 2004.

[5] K. A. Kopotun, On K-monotone interpolation, Advances in Constructive Approximation, Nashboro Press, M. Neamtu and E. B. Saff (eds.), 265-275, 2004.