Comparison between some techniques of interpolators: An application in engineering

Comparación entre algunas técnicas de interpoladores una aplicación en ingeniería

D. M. Devia-Narváez; G. Correa-Vélez; F. Mesa

***Abstract*— This paper contains a detailed comparison between the Lagrange and Hermite polynomial interpolation when approximating a function. Initially the theoretical part about each interpolation procedure is exposed, followed by the exposure of the conditions of the comparative analysis between approximations, explaining the tests and calculating the error to be made.**

***Index Terms*— Interpolation, approximation, polynomial, function, Lagrange, Hermite.**

***Resumen*— El presente artículo contiene una detallada comparación entre la interpolación polinómica de Lagrange y de Hermite al momento de aproximar una función. Inicialmente se exponen la parte teórica acerca de cada procedimiento de interpolación, se exponen las condiciones de análisis comparativo entre aproximaciones, explicando las pruebas y el cálculo del error a realizar.**

***Palabras claves*— Interpolación, aproximación, polinomio, función, Lagrange, Hermite.**

I. INTRODUCTION

IN the mathematical field of numerical analysis, polynomial interpolation is a method used to know, in an approximate way, the values that a certain function takes, of which only its image is known in a finite number of abscissas. Often, the expression of the function will not even be known and only the values it takes for these abscissas will be available. [1]

In numerous phenomena of nature we observe a certain regularity in the way of producing, this allows us to draw conclusions from the progress of a phenomenon in situations that we have not measured directly. [2]

Interpolation consists of finding a data within a range in which we know the values at the extremes. In engineering and some sciences, it is common to have a certain number of points obtained by sampling or from an experiment and to try to build a function that adjusts them.

Another problem closely linked with that of interpolation is the approximation of a function complicated by a simpler one. By having a function whose calculation is costly, you can start from a certain number of its values and interpolate these data by constructing a simpler function (interpolating polynomial). In general, of course, the same values are not obtained by evaluating the function obtained than if the original function is evaluated, although depending on the characteristics of the problem and the method of interpolation used, the gain in efficiency can compensate for the error committed. [2]

The general problem of interpolation is presented to us when we are given a function of which we only know a series of points of it and we are asked to find the value of a point of this function.

It is desired, therefore, to find a function whose graph passes through these points and which serves to estimate the desired values.

The treatment for both problems is similar; the "interpolator" polynomials will be used. One of the best-known applications of these methods is for laboratory data, disease reports or the probability that a natural and non-natural phenomenon will occur. [3]

Returning to the previous ideas, the objective is to find a polynomial that fulfills the aforementioned and that allows to find approximations of other unknown values.e.

II. CONTENT

A. Taylor polynomial

Suppose that f (x) admits continuous derivatives of all orders in a range (a, b) in which the point x0 is. Suppose that the Taylor polynomial sequence converges to f (x). [4]

The great difficulty of the construction of the previous definition is the calculation of the higher derivatives since in

This manuscript was sent on December 15, 2017 and accepted on March 15, 2019

D. M. Devia-Narváez, Universidad Tecnológica de Pereira and GIMAE (Grupo de Investigación en Matemática Aplicada y Educación). Pereira, Colombia. Correo-e:dmdevian@utp.edu.co

G. Correa-Vélez, Universidad Tecnológica de Pereira and GIMAE (Grupo de Investigación en Matemática Aplicada y Educación). Pereira, Colombia. Correo-e: gecove@utp.edu.co

F. Mesa, Universidad Tecnológica de Pereira and GIMAE (Grupo de Investigación en Matemática Aplicada y Educación). Pereira, Colombia. Correo-e: femesa@utp.edu.co

many occasions these are difficult to find. [5]

Another way many more practical is to suppose n + 1 point (x_i, y_i) where the components x_i are distributed over the domain of the function and thus avoid the calculation of the higher derivatives, as it is intended to build a function that passes through each of them. these points what the method guarantees whatever is used should be the calculation of the errors, that is, this should be minimized.

B. Lagrange interpolation polynomial

The Lagrange polynomial interpolation is the way to present the polynomial that interpolates a given set of points under the following conditions: [4] [6] [7]

Let the set of points (x_i, y_i) with i = 1,2,3, ..., n, and assuming all the different x_i, the Lagrange interpolation polynomial is defined as the linear combination [1]: $P(x) = \sum_{i=1}^{n} y_i \, L_i(x)$

$$L_i(x) = \prod_{\substack{i=1 \\ i \neq k}}^{n} \frac{x - x_k}{x_i - x_k} \qquad (1)$$

Lagrange Interpolator Code

```
clear all
clc
syms x
n=input('Enter data number to
interpolate: ');
for i=1:n
    xi(i)=input(['Enter values of
x(',num2str(i),'):']);
  end

for j=1:n
    y(j)=input(['Enter values of
y(',num2str(j),'):']);
end

Ln=0;
for k=1:n
    L=1;
    for i=1:n
        if k==i
            L=L*1;
        else
          L= L*(x-xi(i))/(xi(k)-xi(i));
        end
    end
    Ln=Ln+L*y(k);
    clear L
end

disp('El polinomio producto de la
interpolacion de Lagrange es: ');
Ln=expand(Ln);
disp(expand(Ln));
hold on
ezplot(Ln, [min(xi)-2 max(xi)+2])
title('Interpolacion de Lagrange');
hold on
plot(xi,y,'rx')
```

C. Hermite interpolation polynomial

Hermite polynomial interpolation generates a polynomial that approximates the function f in the interval [a, b] in xi for i = 0, 1,.. n, besides this interpolating polynomial fulfills the condition that the value of the evaluation of the point xi in its first derivative coincides with that of f, in such a way that: [8]

$$\frac{d}{dx}P(x_i) = \frac{d}{dx}f(x_i) \qquad para\ cada\ i = 0,1,2,\dots,n$$

The Hermite polynomial generated has a great relationship with Newton's polynomial, while both share the calculation of divided differences. The Hermite interpolation polynomial has the form: [9]

$$P(x) = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1) + \cdots + b_n(x - x_0)(x - x_1)\cdots(x - x_{n-1})$$

where:

$$b_0 = f[z_0] = f(b_0)$$
$$b_1 = f[z_0, z_1] = f'(b_0)$$
$$b_2 = f[z_0, z_1, z_2] = \frac{f[z_1, z_2] - f[z_0, z_1]}{z_1 - z_0}$$
$$\vdots$$
$$b_n = f[z_0, z_1, \dots, z_n] = \frac{f[z_1, z_2, \dots, z_n] - f[z_0, z_1, \dots, z_{n-1}]}{z_n - z_0}$$

These coefficients are calculated according to the table of divided differences (Table 1).

TABLE 1
TABLE OF DIVIDED DIFFERENCES

z	$f(z)$	Primeras diferencias divididas	Segundas diferencias divididas
$Z_0 = X_0$	$f[Z_0] = f(X_0)$		
		$f[z_0, z_1] = f'(x_0)$	
$Z_1 = X_0$	$f[Z_1] = f(X_0)$		$f[z_0, z_1, z_2] = \frac{f[z_1, z_2] - f[z_0, z_1]}{z_2 - z_0}$
		$f[z_1, z_2] = \frac{f[z_2] - f[z_1]}{z_2 - z_1}$	
$Z_2 = X_1$	$f[Z_2] = f(X_1)$		$f[z_1, z_2, z_3] = \frac{f[z_2, z_3] - f[z_1, z_2]}{z_3 - z_1}$
		$f[z_2, z_3] = f'(x_1)$	
$Z_3 = X_1$	$f[Z_3] = f(X_1)$		$f[z_2, z_3, z_4] = \frac{f[z_3, z_4] - f[z_2, z_3]}{z_4 - z_2}$
		$f[z_3, z_4] = \frac{f[z_4] - f[z_3]}{z_4 - z_3}$	
$Z_4 = X_2$	$f[Z_4] = f(X_2)$		$f[z_3, z_4, z_5] = \frac{f[z_4, z_5] - f[z_3, z_4]}{z_5 - z_3}$
		$f[z_4, z_5] = f'(x_2)$	
$Z_5 = X_2$	$f[Z_5] = f(X_2)$		

D. Test function to approximate

To establish a clear difference between the approximation of the interpolating polynomials, we will take test functions and some values of the x axis that will provide the points for the respective interpolation. [9]

Hermite interpolation code

```
clear all
clc
syms x
p=input('Enter data number to
interpolate: ')
n=p
for i=1:n
    xi(i)=input(['Enter values of
x(',num2str(i),'):']);
  end

for j=1:n
    y(j)=input(['Enter values of
 y(',num2str(j),'):']);
end
for j=1:n
    yp(j)=input(['Enter values of
dy/dx(',num2str(j),'):']);
end
n=2*p
a=zeros(n,n+1);
for j=1:p
a(2*j-1,1:2)=[xi(j) y(j)]
a(2*j,1:2)=[xi(j) y(j)]
b(2*j-1,1)=[yp(j)]
   end
   k=3;

   for i=1:n-1
       if rem(i,2)==1
               a(i,k)=b(i);
       else
           a(i,k)=(a(i+1,k-1)-a(i,k-
   1))/(a(i-2+k,1)-a(i,1))
       end
   end
   clear k;


   for k=4:n+1
       if k<n+1
       for i=1:n-1
           if i-2+k>n
               a(i,k)=0
           else
           a(i,k)=(a(i+1,k-1)-a(i,k-
   1))/(a(i-2+k,1)-a(i,1))
           end
       end

       else
           a(1,k)=(a(2,k-1)-a(1,k-
1))/(a(1+k-2,1)-a(1,1))
    end
end


Ln=0;
for k=2:n+1
    L=a(1,k)
    if k==2
     L=L;
    elseif k>2
    for i=1:k-2
            L=L*(x-a(i,1))
    end
    end
    Ln=Ln+L
    clear L
end

hold on
Pn=expand(Ln)
ezplot(Pn,[min(xi)-2 max(xi)+2])
hold on
plot(xi,y,'rx')
```

The functions that will be put as test are the exponential function, the sinusoidal function and the polynomial function.

E. *Exponential test function*

$$f(x) = e^x$$

TABLE2
EXPONENTIAL TEST FUNTION DATA TABLE

i	x_i	$f(x_i)$	$f'(x_i)$
1	0	1	1
2	2	e^2	e^2
3	4	e^4	e^4
4	6	e^6	e^6
5	8	e^8	e^8

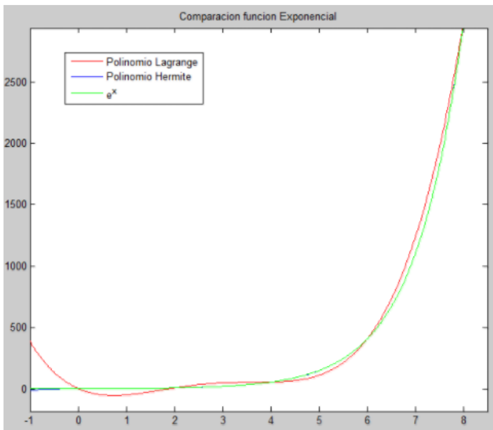

Fig 1. Exponential function

F. *Sinusoidal test function*

$$f(x) = sen(x)$$

TABLE 3
DATA TABLE SINOSOIDAL TEST FUNCTION

i	x_i	$f(x_i)$	$f'(x_i)$
1	0	0	1
2	$\pi/2$	1	0
3	π	0	-1
4	$3\pi/2$	-1	0
5	2π	0	1

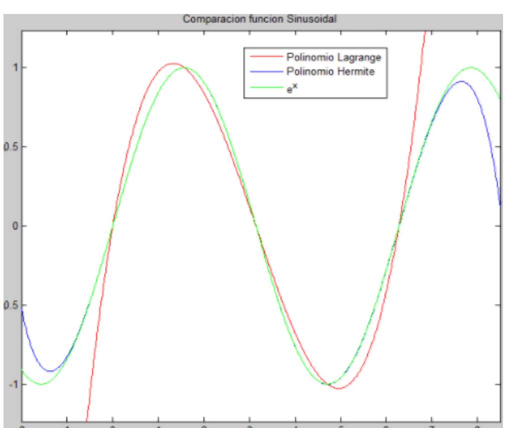

Fig. 2. Graph of the sinusoidal function

Polynomial test function

$$f(x) = \frac{1}{4}x^4 + \frac{1}{2}x^3 - x^2 - x + \frac{1}{2}$$

TABLE 4
POLYNOMIAL TEST FUNCTION DATA TABLE

i	x_i	$f(x_i)$	$f'(x_i)$
1	-1	$1/4$	$3/2$
2	0	$1/2$	-1
3	1	$-3/4$	$-1/2$

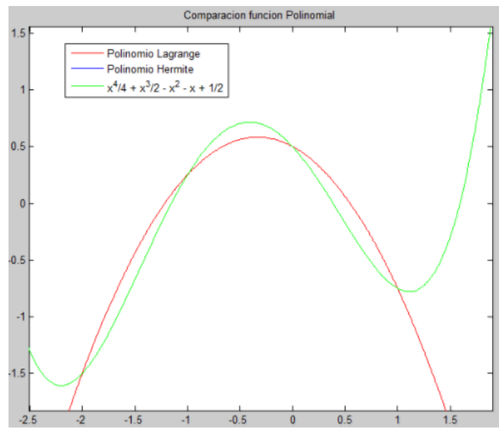

Fig. .3 Graph of the polynomial function

G. *Calculation of the error*

The measure of the mean square error (ECM) is established to properly observe the difference between the test function and the approximation of the interpolator polynomial [10]

$$ECM = \sum_{i=1}^{n-1} [f(x_i^*) - p(x_i^*)]^2,$$

$$x_i^* = x_0 + i\frac{\Delta x}{2}, \qquad \Delta x = \frac{x_i - x_{i-1}}{n} \tag{2}$$

As it is specified in (2) will be taken as error evaluation values (abscissas) those that are intermediate to the interpolation points, that is, have n points to generate p (x) and n-1 points for evaluate the error. [11], [12], [13]

III. RESULTS

The results obtained will be presented for different test functions, exponential, sinusoidal and polynomial functions, these were chosen, because the vast majority of engineering functions can be expressed as a combination of those already mentioned.

A. *Exponential test function:*

B. *1 case. Lagrange polynomial*

$$p(x) = 4.339x^4 - 46.638x^3 + 163.43x^3$$

$$-171.828x + 1.0$$

C. 2 case. Hermite polynomial

$$\begin{aligned} p(x) = 0.000215667x^9 \, - \, 0.00566922x^8 \\ + \, 0.0671179x^7 \, - \, 0.433464x^6 \\ + \, 1.67294x^5 \, - \, 3.67882x^4 \\ + \, 4.63325x^3 \, - \, 1.714622 + \, x \, + \, 1.0 \end{aligned}$$

D. Sinusoidal test function:

$$f(x) = senx$$

E. Case 1. Lagrange polynomial

$$p(x) = \frac{8}{3\pi^3}x^3 - \frac{8}{\pi^2}x^2 + \frac{16}{3\pi}x$$

F. Case 2. Hermite polynomial

$$\begin{aligned} p(x) = -2.1989 \times 10^{-6}x^9 \, + \, 0.00006.2173 \times 10^{-5}x^8 \\ - \, 5.8726 \times 10^{-3}x^7 + \, 1.4602 \times 10^{-3}x^6 \\ + \, 0.0049058x^5 + \, 0.0049190x^4 \\ - \, 0.1706x^3 \, + \, 0.001344x^2 \, + \, x \end{aligned}$$

G. Polynomial test function:

H. Case 1. Lagrange polynomial

$$p(x) = -\frac{3}{4}x^2 - \frac{1}{2}x + \frac{1}{2}$$

I. Case 2. Hermite polynomial

$$f(x) = p(x) = \frac{1}{4}x^4 + \frac{1}{2}x^3 - x^2 - x + \frac{1}{2}$$

J. Mean square error

TABLE 5
DATA TABLE ECM

Función/ Polinomio interpolador	**Exponencial**	**Sinusoidal**	**Polinómica**
Lagrange	5582.211	0.0174648	0.0186767
Hermite	0.0313526	1.148×10^{-9}	0

IV. CONCLUSION

The results obtained from the interpolating polynomials of Lagrange and Hermite quickly show that the Hermite polynomial will always be of greater degree than the Lagrange polynomial, a fact that represents a greater use of computational memory, however, in Table 5 of the EMC, the efficiency of this computational cost, which is a result of zero error and almost zero in some approximations made.

Finally, we can see on the graphs, the Hermite interpolator is more reliable than the Lagrange interpolator, almost unlike the original function, due to the information required to perform the interpolation.

REFERENCES

[1] R.L. Burden, and J.D. Faires, “Numerical Analysis”, 2.1 - The Bisection Method, Brooks/Cole, Cengage Learning, 2011.
[2] M. Crouzeix and A. L. “Mignot, Analyse numérique des équations différentielles” 2nd, Masson, Parıs, 1992.
[3] D. Kincaid, and W. Cheney, “Numerical analysis the mathematics of scientific calculation”, (No. 511.7 K5Y) (1994).
[4] C. A. Schloeder, and N. E. Zimmerman, and M. J. Jacobs. "Comparison of methods for interpolating soil properties using limited data." Soil Science Society of America Journal 65.2 (2001): 470-479.
[5] J. H. Mathews, and K. D. Fink, “Métodos numéricos con Matlab”. Pearson, España (2000).
[6] Witenberg, J. P. “Methods and models of operations research” (Vol. 1). Editorial Limusa (1999).
[7] S. Nakamura, “Métodos Numéricos Aplicados con Software”. Prentice-Hall Hispanoamericana, S.A. México 1992.
[8] E. Cheney, Introduction to Approximation Theory. Internat. Ser. Pure ans Applied Mathematics. McGraw-Hill. 1966.
[9] K. Atkinson, “An Introduction to Numerical Analysis”, John Wiley, New York (1989).
[10] B. Carnahan, H. Luther, and Wilkes J. Applied Numerical Methods. John Wiley ans Sons, Inc., New York (1969).
[11] J. Demmel “Applied Numerical Linear Algebra”. SIAM, Philadelphia (1997).
[12] A. Quarteroni, R. Sacco, and F. Saleri, “Numerical Mathematics,of Texts in Applied Mathematics”, Springer-Verlag, New York, 2nd edition (2006).
[13] C. Uberhuber, “Numerical Computation: Methods, Software, and Analysis”, Springer-Verlag, Berlin Heidelberg (1997).

Diana Marcela Devia Narváez. Associated teacher of mathematics and physics at Universidad Tecnologica de Pereira, Colombia. PhD in engineering (2012). Master of Science (Msc) in the faculty of Physics - Cience (2010). Member of plasma laboratory from Universidad Nacional de Colombia located in Manizales, Colombia. Member of nonlinear differential equations group “GEDNOL” at Universidad Tecnologica de Pereira, Colombia. Area of expertise: material processing through assisted plasma techniques, structural characterization mechanical of materials, simulation and modeling of material’s physical properties.

German Correa Vélez, university lecturer of the department of Mathematics at the Universidad Tecnológica de Pereira, Colombia. Master of Science (Msc) in mathematics (2008). Member of nonlinear differential equations group “GEDNOL” at Universidad Tecnologica de Pereira, Colombia. Member of the research group in Applied Mathematics and Education “GIMAE” at the Universidad Tecnologica de Pereira.

Fernado Mesa, university professor and director of the Mathematics Department of the Technological University of Pereira, Colombia. Master of Science (Msc) in physical instrumentation (2007). Member of the plasma laboratory of the National University of Colombia, located in Manizales, Colombia. Member of the group of non-linear differential equations "GEDNOL" at the Technological University of Pereira, Colombia. Director of the research group in Applied Mathematics and Education "GIMAE" at the Universidad Tecnologica de Pereira, Colombia.