Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.18576/amis/140120

Newton's P-Difference Interpolation Formula for Interval-Valued Function

Md Sadikur Rahman, Md Akhtar, Ali Akbar Shaikh and Asoke Kumar Bhunia*

Department of mathematics, The University of Burdwan, Burdwan, India

Received: 21 Feb. 2019, Revised: 22 Mar. 2019, Accepted: 24 Mar. 2019.
Published online: 1 Jan. 2020.

Abstract: Interpolation formula is an important concept in the theory of numerical analysis which is grown up based on interpolation. So, to study the interpolation in interval environment, interval interpolation formulae are more essential. The objective of this article is to establish extended Newton's interpolation formulae for interval-valued functions using p-difference of intervals. For this purpose, parametric representation of intervals with interval arithmetic in the parametric form and parametric representation of interval-valued function have been discussed briefly. Using p-difference of intervals, finite differences (forward/backward) of interval-valued function have been defined and called them as Newton's p-difference (forward/backward) operators. After that fundamental theorem of finite difference calculus has been extended for polynomials with interval-valued coefficients. Then Newton's p-difference interpolation formulae (forward/backward) have been derived. Also, computational algorithm for forward p-difference interpolation has been established. Finally, with the help of graphical representation of a numerical example, it has been shown that both the p-difference interpolation formulae (forward and backward) are identical. It should be noted that the proposed interpolation formulae are the generalization of traditional Newton's interpolation formulae (forward and backward).
Keywords: Interpolation; p-difference; Newton's p-difference interpolation formula; parametric representation.

1 Introduction

Interpolation has a crucial impact in all the fields of science, commerce and technology for research works based on numerical data. It is a process of approximation for obtaining a simple function from a given set of data points in a certain condition that the function satisfies the given data points. Basically, interpolation is used to approximate the complicated functions (either its derivative at a point cannot be determined or its roots cannot be determined or its integration cannot be found in certain region) into a simple function. Since polynomial is easily differentiable, integrable and easier to find the roots, it is commonly used for interpolation and this interpolation is called as polynomial interpolation. There are several interpolation formulae such as Newton's interpolation formulae (forward/backward), Lagrange interpolation formula etc., for fixed data points in the existing literature [Akima [1], Kincaid and Cheney [2], Gerald and Wheatly [3], Barrault *et al.* [4] and others].

However, in reality, there are several situations (e.g., data of temperature for different days in a week, data of rain fall in a city of successive 10 years etc.) where given data points are not fixed due to uncertainty. In these uncertain situations, flexible data points can be represented precisely as either a fuzzy number or a fuzzy set with proper membership function (Huang and Shen [5]) or a random variable with appropriate distribution function or an interval number (Moore [6], Moore *et al.* [7]). Among these representations, interval representation is easier and efficient representation. An interval $A=[a_L,a_U]$ is represented in the following parametric forms (Ramezanadeh *et al.* [8]):

(i) $A=\{a(s):a(s)=a_L+s(a_U-a_L),s\in[0,1]\}$
(increasing form)

(ii) $A=\{a(s):a(s)=a_U-s(a_U-a_L),s\in[0,1]\}$
(decreasing form)

Now, the process of interpolation with interval-valued data points is called interval interpolation (Markov [9]). It is

*Corresponding author e-mail: aliashaikh@math.buruniv.ac.in, aakbarshaikh@gmail.com

important to know how interval data involving in different real-life-problems are manipulated or estimated. So, the formula for interval interpolation is essential and derivation of interval interpolation formulae is an interesting research topic in the area of numerical analysis.

This concept of interpolation in a crisp or interval environment have many applications in the different area of physical system such as, theory of non-linear dynamical system, oscillation theory, application of factional calculus and fractals theory. In this context, some recent works are reported here: Ali et al. [10] studied the generalized formula of Hirota-Satusma coupled KDV system. Owyed et al. [11] established a new optical solution of evolution equation in optics theory. In the same year Owyed et al. [12] also contributed their work in fractional theory. Elgendy et al. [13] used the concept of interpolation in the physical chemistry and in the same year Ismail et al. [14] worked on fractional oscillation theory.

In this work, we have generalized Newton's interpolation formula for interval-valued functions using finite p-differences (forward/backward). For this purpose, first of all, we have discussed parametric representation of intervals with interval arithmetic in the parametric form and parametric representations of interval-valued function briefly. After that we have extended the fundamental theorem of finite difference calculus in interval environment. Then we have derived the main result of this article: Newton's p-differences interpolation formulae (forward/backward). Finally, with the help of graphical representation of numerical example, we have shown that Newton's p-difference interpolation formulae (forward and backward) are identical.

2 Basic Concepts (Ramezanadeh *et al.* [8])

In this section, we have discussed the definitions of arithmetic operations of intervals in parametric form (increasing/decreasing). Suppose K_c be the set of nonempty closed-bounded intervals of $\sim$.

Let $A=[a_L,a_U]\in K_c$, then the parametric representations of A are as follows:

(i) $A=\{a(s):a(s)=a_L+s(a_U-a_L),s\in[0,1]\}$ (Increasing Form or IF),

(ii) $A=\{a(s):a(s)=a_U-s(a_U-a_L),s\in[0,1]\}$ (Decreasing Form or DF).

Definition1 Let $\{a(s):s\in[0,1]\}$ and $\{b(s):s\in[0,1]\}$ be the IFs (DFs) of $A=[a_L,a_U]$ and $B=[b_L,b_U]$ respectively, and λ be a real number. The parametric arithmetic in K_c can be defined as

$$(a)\,A+B=\{a(s_1)+b(s_2):s_1,s_2\in[0,1]\},$$
$$(b)\,A-B=\{a(s_1)-b(s_2):s_1,s_2\in[0,1]\},$$
$$(c)\,A.B=\{a(s_1)b(s_2):s_1,s_2\in[0,1]\},$$
$$(d)\,A/B=\{a(s_1)/b(s_2):b(s_2)\neq 0,s_1,s_2\in[0,1]\},$$
$$(e)\,\lambda.A=\{\lambda a(s_1):s_1\in[0,1]\},$$
$$(f)\,A(-)_pB=\{a(s)-b(s):s\in[0,1]\},$$
$$(g)\,A=B\Leftrightarrow\{a(s):s\in[0,1]\}=\{b(s):s\in[0,1]\}.$$

2.1 Parametric Form of Interval-Valued Function (IVF)

Let $G:D\subseteq\sim\to K_c$ be an IVF with lower-upper bound form, $G(x)=[g_L(x),g_U(x)]$. Then the parametric representations of $G(x)$ are

$$(i)\{g_L(x)+s(g_U(x)-g_L(x)):s\in[0,1]\},$$
$$(ii)\{g_U(x)-s(g_U(x)-g_L(x)):s\in[0,1]\}.$$

An IVF with real independent variable and interval coefficients is a special case of $G:D\subseteq\sim\to K_c$. Using IRs of coefficients, it is possible to write this function as a set of crisp functions, an alternative parametric representation of $G(x)$ as follows:

$$G(x)=\left\{\begin{array}{l} g(x,s):s=(s_1,s_2,\dots,s_k)\in[0,1]^l, \\ \text{where } l \text{ is number of interval coefficients involves in } G(x)\end{array}\right\}$$

2.2 Polynomial with Interval Coefficients

An interval-valued function $P_k:D\subseteq\sim\to K_c$ of the form

$$P_k(x)=[a_{0L},a_{0U}]+[a_{1L},a_{1U}]x+[a_{2L},a_{2U}]x^2$$
$$+\dots+[a_{kL},a_{kU}]x^k,\ 0\notin[a_{kL},a_{kU}],$$

is called a polynomial of degree k with interval coefficients and the alternative parametric representation of P_k is

$$P_k(x)=\left\{\begin{array}{l} a_0(s_0)+a_1(s_1)x+a_2(s_2)x^2 \\ +...+a_k(s_k)x^k : s_i\in[0,1], \\ i=1,2,...,k \text{ and } a_k(s_k)\neq 0, \forall s_k \end{array}\right\}$$

where

$a_i(s_i)=a_{iL}+s_i(a_{iU}-a_{iL}), s_i\in[0,1]$ and $i=1,2,...,k.$

Proposition 1 Let $G:D\subseteq^{\sim}\to K_c$ be an IVF in parametric representation form

$$G(x)=\left\{\begin{array}{l} g(x,s):s=(s_1,...,s_l)\in[0,1]^l, \\ \text{where } l \text{ is number of interval} \\ \text{coefficients involves in } G(x) \end{array}\right\}.$$

Then $G(x)$ can be represented in the lower-upper bound form as: $G(x)=[g_L(x),g_U(x)].$

where

$$g_L(x)=\min_s\left\{\begin{array}{l} g(x,s):s=(s_1,...,s_l)\in[0,1]^l, \\ \text{where } l \text{ is number of interval} \\ \text{coefficients involves in } \mathrm{G}(x) \end{array}\right\},$$

$$f_U(x)=\max_s\left\{\begin{array}{l} g(x,s):s=(s_1,...,s_l)\in[0,1]^l, \\ \text{where } l \text{ is number of interval} \\ \text{coefficients involves in } \mathrm{G}(x) \end{array}\right\}.$$

3 Main Results

In this section we have discussed the main contribution of the work- Newton p-difference interpolation formulae (forward/backward) for interval-valued data points. In this purpose we have first defined finite p-differences (forward/backward) and extend the fundamental theorem of difference calculus into p-difference calculus.

3.1 Finite P-differences Operators

Let $Y=[y_L,y_U]=[g_L(x),g_U(x)]$ be an IVF defined on $[a,b]$. The points $x_0,x_1,\dots,x_k$ are taken from in such a way such that $x_i=x_0+ih, i=0,1,\dots,k.$ Let $Y_i=[y_{iL},y_{iU}]=[g_L(x_i),g_U(x_i)].$

2.3.1 Forward P-difference

The forward p-difference of an IVF be defined as

$$\Delta_p[g_L(x),g_U(x)]=[g_L(x+h),g_U(x+h)]$$
$$(-)_p[g_L(x),g_U(x)]$$

In terms of $Y_i=[y_{iL},y_{iU}]$ at $x=x_i$, the above relation gives

$$\Delta_p[g_L(x_i),g_U(x_i)]=[g_L(x_i+h),g_U(x_i+h)]$$
$$(-)_p[g_L(x_i),g_U(x_i)]$$

i.e., $\Delta_{\mathrm{p}}[y_{iL},y_{iU}]=[y_{(i+1)L},y_{(i+1)U}](-)_p[y_{iL},y_{iU}]$

Therefore,

$$\Delta_{\mathrm{p}}[y_{0L},y_{0U}]=[y_{1L},y_{1U}](-)_p[y_{0L},y_{0U}]$$
$$\Delta_{\mathrm{p}}[y_{1L},y_{1U}]=[y_{2L},y_{2U}](-)_p[y_{1L},y_{1U}]$$

and so on,

$$\Delta_{\mathrm{p}}[y_{(k-1)L},y_{(k-1)U}]=[y_{kL},y_{kU}](-)_p[y_{(k-1)L},y_{(k-1)U}].$$

The p-differences of first order p-differences (forward) are called second order p-differences (forward) and they are denoted by $\Delta_p^2Y_0,\Delta_p^2Y_1,...$

In a similar manner, higher p-differences can be defined.

Proposition 2 Let

$[g_L(x),g_U(x)]=\{g(x,s):s\in[0,1]\}$ be an IVF defined on $[a,b]$ then forward p-difference of $[g_L(x),g_U(x)]$ is of the following form:

$$\Delta_p[g_L(x),g_U(x)]=\{\Delta g(x,s):s\in[0,1]\}.$$

Proof From the definition of forward p-difference we can say,

$$\Delta_p[g_L(x),g_U(x)]=[g_L(x+h),g_U(x+h)] \quad (1)$$
$$(-)_p[g_L(x),g_U(x)]$$

Now, using the parametric representation of an interval, we can re-write (1) as,

$$\Delta_p[g_L(x),g_U(x)]=\{g(x+h,s)-g(x,s):s\in[0,1]\}$$
$$=\{\Delta g(x,s):s\in[0,1]\}$$

(using the forward difference)

Remark1

$$(i)\,\Delta_p^2 Y_0=\Delta_p Y_1 (-)_p \Delta_p Y_0=\left\{\begin{array}{l}\Delta y_1(s)-\Delta y_0(s)\\ :s\in[0,1]\end{array}\right\}$$

$$=\{y_2(s)-2y_1(s)+y_0(s):s\in[0,1]\}$$

and similarly, $\Delta_p^2 Y_1=\left\{\begin{array}{l}y_3(s)-2y_2(s)+y_1(s)\\ :s\in[0,1]\end{array}\right\}$

$$(ii)\,\Delta_p^3 Y_0=\Delta_p^2 Y_1 (-)_p \Delta_p^2 Y_0=\left\{\begin{array}{l}\Delta^2 y_1(s)-\Delta^2 y_0(s)\\ :s\in[0,1]\end{array}\right\}$$

$$=\{y_3(s)-3y_2(s)+3y_1(s)-y_0(s):s\in[0,1]\}$$

Forward p-difference table for interval-valued data and forward difference table for corresponding parametric data of interval-valued data are given in Table 1.

Table 1: Forward p-difference table.

x	Y	$\Delta_p Y$	$\Delta_p^2 Y$	$\Delta_p^3 Y$	$\Delta_p^4 Y$
x_0	Y_0				
		$\Delta_p Y_0$			
x_1	Y_1		$\Delta_p^2 Y_0$		
		$\Delta_p Y_1$		$\Delta_p^3 Y_0$	
x_2	Y_2		$\Delta_p^2 Y_1$		$\Delta_p^4 Y_0$
		$\Delta_p Y_2$		$\Delta_p^3 Y_1$	
x_3	Y_3		$\Delta_p^2 Y_2$		
		$\Delta_p Y_3$			
x_4	Y_4				

Using **Definition 1(g)**, for all $s\in[0,1]$, the above forward p-difference table has been converted into the forward difference table (cf. Table 2):

Backward P-difference

The backward p-difference of an IVF is defined as follows:

$$\nabla_p[g_L(x),g_U(x)]=[g_L(x),g_U(x)]$$
$$(-)_p[g_L(x-h),g_U(x-h)]$$

In terms of $Y_i=[y_{iL},y_{iU}]$ at $x=x_i$, the above relation gives

Table 2: Forward difference table in parametric form.

x	$y(s)$	$\Delta y(s)$	$\Delta^2 y(s)$	$\Delta^3 y(s)$	$\Delta^4 y(s)$
x_0	$y_0(s)$				
		$\Delta y_0(s)$			
x_1	$y_1(s)$		$\Delta^2 y_0(s)$		
		$\Delta y_1(s)$		$\Delta^3 y_0(s)$	
x_2	$y_2(s)$		$\Delta^2 y_1(s)$		$\Delta^4 y_0(s)$
		$\Delta y_2(s)$		$\Delta^3 y_1(s)$	
x_3	$y_3(s)$		$\Delta^2 y_2(s)$		
		$\Delta y_3(s)$			
x_4	$y_4(s)$				

Backward P-difference

The backward p-difference of an IVF is defined as follows:

$$\nabla_p[g_L(x),g_U(x)]=[g_L(x),g_U(x)]$$
$$(-)_p[g_L(x-h),g_U(x-h)]$$

In terms of $Y_i=[y_{iL},y_{iU}]$ at $x=x_i$, the above relation gives

$$\nabla_p[g_L(x_i),g_U(x_i)]$$
$$=[g_L(x_i),g_U(x_i)](-)_p[g_L(x_i-h),g_U(x_i-h)]$$

i.e., $\nabla_p[y_{iL},y_{iU}]=[y_{iL},y_{iU}](-)_p[y_{(i-1)L},y_{(i-1)U}]$

Therefore,

$$\nabla_p[y_{1L},y_{1U}]=[y_{1L},y_{1U}](-)_p[y_{0L},y_{0U}];$$
$$\nabla_p[y_{2L},y_{2U}]=[y_{2L},y_{2U}](-)_p[y_{1L},y_{1U}];$$

and so on,

$$\nabla_p[y_{kL},y_{kU}]=[y_{kL},y_{kU}](-)_p[y_{(k-1)L},y_{(k-1)U}].$$

The p-differences of the first order p-differences (backward) are called second order p-differences (backward) and they are denoted by $\nabla_p^2 Y_1,\nabla_p^2 Y_2\ldots$

Similarly, the higher order p-differences (backward) can be defined.

Proposition 3:

Let $\left[g_L(x), g_U(x)\right] = \{g(x,s) : s \in [0,1]\}$ be an IVF defined on $[a,b]$ then backward p-difference of $\left[g_L(x), g_U(x)\right]$ is of the following form:

$$\nabla_p \left[g_L(x), g_U(x)\right] = \{\nabla g(r,s) : s \in [0,1]\}.$$

Proof From the definition of forward p-difference we can say,

$$\nabla_p \left[g_L(x), g_U(x)\right] = \left[g_L(x), g_U(x)\right] (-)_p \left[g_L(x-h), g_U(x-h)\right] \quad (2)$$

Now, using the parametric representation of an interval, we can re-write (2) as,

$$\begin{aligned}\nabla_p \left[g_L(x), g_U(x)\right] &= \{g(x,s) - g(x-h,s) : s \in [0,1\\ &= \{\nabla g(x,s) : s \in [0,1]\}\end{aligned}$$

(using the backward difference)

Remark 2

$(i)\nabla_p^2 Y_1 = \nabla_p Y_1 (-)_p \nabla_p Y_0 = \{\nabla y_1(s) - \nabla y_0(s) : s \in [0,1]\}$
$= \{y_2(s) - 2y_1(s) + y_0(s) : s \in [0,1]\}$

and similarly, $\nabla_p^2 Y_2 = \{y_3(s) - 2y_2(s) + y_1(s) : s \in [0,1]\}$

$(ii)\nabla_p^3 Y_1 = \nabla_p^2 Y_1 (-)_p \nabla_p^2 Y_0 = \{\nabla^2 y_1(s) - \nabla^2 y_0(s) : s \in [0,1]$
$= \{y_3(s) - 3y_2(s) + 3y_1(s) - y_0(s) : s \in [0,1]\}$

Backward p-difference table for interval-valued data and backward difference for corresponding parametric data of interval-valued data are given in the Table 3.

Table 3: Backward p-difference table.

x	Y	$\nabla_p Y$	$\nabla_p^2 Y$	$\nabla_p^3 Y$	$\nabla_p^4 Y$
x_0	Y_0				
x_1	Y_1	$\nabla_p Y_1$			
x_2	Y_2	$\nabla_p Y_2$	$\nabla_p^2 Y_2$		
x_3	Y_3	$\nabla_p Y_3$	$\nabla_p^2 Y_3$	$\nabla_p^3 Y_3$	
x_4	Y_4	$\nabla_p Y_4$	$\nabla_p^2 Y_4$	$\nabla_p^3 Y_4$	$\nabla_p^4 Y_4$

Using **Definition 1(g)**, for all $s \in [0,1]$, the above backward p-difference table has been converted into the backward difference table (cf. Table 4):

Table 4: Backward difference table in parametric form.

x	$y(s)$	$\nabla y(s)$	$\nabla^2 y(s)$	$\nabla^3 y(s)$	$\nabla^4 y(s)$
x_0	$y_0(s)$				
x_1	$y_1(s)$	$\nabla y_1(s)$			
x_2	$y_2(s)$	$\nabla y_2(s)$	$\nabla^2 y_2(s)$		
x_3	$y_3(s)$	$\nabla y_3(s)$	$\nabla^2 y_3(s)$	$\nabla^3 y_3(s)$	
x_4	$y_4(s)$	$\nabla y_4(s)$	$\nabla^2 y_4(s)$	$\nabla^3 y_4(s)$	$\nabla^4 y_4(s$

Theorem 1 (Fundamental theorem of p-difference calculus)

The k^{th} p-difference of a polynomial *G(x)* with interval coefficient of degree '*k*' is constant interval and $(k+1)^{th}$ p-difference vanishes.

Proof

Let
$G(x) = A_0 x^k + A_1 x^{k-1} + A_2 x^{k-2} + \ldots + A_{k-1} x + A_k$,
be a polynomial of degree k with interval coefficients.
where $A_i = [a_{iL}, a_{iU}], i = 1,2,\ldots,k.$

Now, $F(x)$ can be written in the parametric form as follows

$$G(x) = \left\{\begin{array}{l} g(x,s) = a_0(s_0)x^k + a_1(s_1)x^{k-1} + a_2(s_2)x^{k-2} \\ +\ldots + a_{k-1}(s_{k-1})x + a_k(s_k) : s_i \in [0,1], i = 0,1,\ldots,k \end{array}\right\}$$

where $a_i(s_i) = a_{iL} + s_i(a_{iU} - a_{iL}), i = 0,1,\ldots,k.$

Then first p-difference of $G(x)$ is

$$\Delta_p G(x) = G(x+h)(-)_p G(x)$$
$$= \left\{\begin{array}{l} g(x+h,s) - g(x,s) : h \text{ is spacing and} \\ s = (s_i : i = 0,1,\ldots,k) \in [0,1]^{k+1} \end{array}\right\}$$

where $g(x,s) = a_0(s_0)x^k + a_1(s_1)x^{k-1} + a_2(s_2)x^{k-2}$
$+\ldots + a_{k-1}(s_{k-1})x + a_k(s_k), s_i \in [0,1], i = 0,1,\ldots,k.$

$$i.e., \Delta_P G(x) = \left\{ \Delta g(x,s) : s = (s_i : i = 0,1,\ldots,k) \in [0,1]^{k+1} \right\},$$

where

$$\Delta g(x,s) = b_0(s_0) x^{k-1} + b_1(s_1) x^{k-2} + \ldots + b_{k-1}(s_{k-1}),$$

$$s_i \in [0,1], i = 0,1,\ldots,k-1$$

and $b_0(s_0) = a_0(s_0) kh$,

$$b_1(s_1) = \left\{ \begin{matrix} a_0(s_0) \dfrac{k(k-1)}{2!} h^2 \\ +a_1(s_1)(k-1)h \end{matrix} \right\}, \ldots, b_{k-1}(s_{k-1}) = a_{k-1}(s_{k-1}) h.$$

The second p-difference of $G(x)$ is

$$\Delta_p^2 G(x) = \Delta_P G(x+h) (-)_p \Delta_p G(x)$$

$$= \left\{ \begin{matrix} \Delta^2 g(x,s) = \Delta g(x+h,s) - \Delta g(x,s) \\ : h \text{ is spacing and } s = (s_i : i = 0,1,\ldots,k-1) \in [0,1]^k \end{matrix} \right\}$$

$$= \left\{ \begin{matrix} c_0(s_0) x^{k-2} + c_1(s_1) x^{k-3} + \ldots + c_{k-2}(s_{k-2}) \\ : s_i \in [0,1], i = 0,1,\ldots,k-2 \end{matrix} \right\}$$

where $c_0(s_0) = b_0(s_0)(k-1)h$,

$$c_1(s_1) = \frac{1}{2!}(k-1)(k-2) b_0(s_0) h^2$$

$$+(k-2) b_1(s_1) h, \ldots, c_{k-2}(s_{k-2}) = b_{k-2}(s_{k-2}) h.$$

In this manner, one can get

$$\Delta_p^{k-1} G(x) = \left\{ \begin{matrix} \Delta^{k-1} g(x,s) = p_0(s_0) x + p_1(s_1) \\ : s_i \in [0,1], i = 0,1 \end{matrix} \right\}$$

Then, $\Delta_p^k G(x) = \left\{ \Delta^k g(x,s) = p_0(s_0) h : s_0 \in [0,1] \right\}$

$$= \left\{ k! h^k a_0(s_0) : s_0 \in [0,1] \right\}$$

$$= \left[\min_{s_0} k! h^k a_0(s_0), \max_{s_0} k! h^k a_0(s_0) \right]$$

$= \left[k! h^k a_{0L}, k! h^k a_{0U} \right]$, which is a constant interval.

And $\Delta_p^{k+1} G(x) = [0,0]$.

3.2 Newton's P-difference Interpolation Formula

Let $[y_L, y_U] = [g_L(x), g_U(x)]$ be an IVF whose explicit form is unknown but the values of $[y_L, y_U]$ at the equispaced points

$x_0, x_1, \ldots, x_k$, i.e., $[y_{iL}, y_{iU}] = [g_L(x_i), g_U(x_i)]$, $i = 0,1,\ldots,k$ are known. Since $x_0, \ldots, x_k$, are equispaced, $x_i = x_0 + ih, \; i = 0,1,\ldots,k$ h is spacing. Now, it has been constructed a polynomial $\psi(x)$ with interval coefficients of degree less than or equal to k satisfying the conditions

$$[y_{iL}, y_{iU}] (-)_p \psi(x_i) = [0,0], i = 0,1,\ldots,k. \tag{3}$$

3.2.1 Newton's p-difference (forward) Interpolation formula:

To derive Newton's forward p-difference interpolation formula, $\psi(x)$ can be taken as,

$$\begin{aligned} \psi(x) &= A_0 + A_1(x-x_0) \\ &+ A_2(x-x_0)(x-x_1) \\ &+ \ldots + A_n(x-x_0)(x-x_1)\ldots(x-x_{n-1}) \end{aligned}, \tag{4}$$

where coefficients $A_i = [a_{iL}, a_{iU}]$ $(i = 0,1,\ldots,k)$ to be determined.

The parametric form of (3) can be written as

$$\left\{ y_i(s) - \psi(x_i, s) = 0 : s \in [0,1], i = 0,1,\ldots,k \right\} \tag{5}$$

where

$$\psi(x,s) = a_0(s) + a_1(s)(x-x_0)$$

$$+a_2(s)(x-x_0)(x-x_1)$$

$$+\ldots + a_k(s)(x-x_0)(x-x_1)\ldots(x-x_{k-1}),$$

$$A_i = \left\{ \begin{matrix} a_i(s) = a_{iL} + s(a_{iU} - a_{iL}) \\ : s \in [0,1] \end{matrix} \right\}$$

and $y_i(s) = y_{iL} + s(y_{iU} - y_{iL})$.

Since (3) holds for all $s \in [0,1]$, $y_i(s) - \psi(x_i, s) = 0$ (6)

To determine the values of $a_i(s)$'s, substituting $x = x_i, i = 0,1,2,\ldots,k$

Putting $x = x_0$ in $\psi(x,s)$ and using (6) we get,

$$\psi(x_0, s) = a_0(s) \; i.e., a_0(s) = y_0(s).$$

For $x = x_1, \psi(x_1, s) = a_0(s) + a_1(s)(x_1 - x_0)$,

or, $y_1(s) = y_0(s) + a_1(s)h$

or, $a_1(s) = \frac{y_1(s) - y_0(s)}{h} = \frac{\Delta y_0(s)}{h}$

Therefore, $A_1 = \left\{ \begin{array}{l} a_1(s) = \frac{\Delta y_0(s)}{h} \\ : s \in [0,1] \end{array} \right\} = \frac{1}{h} \Delta_p [y_{0L}, y_{0U}]$

For $x = x_2, \psi(x_2, s) = a_0(s)$

$+a_1(s)(x_2 - x_0) + a_2(s)(x_2 - x_0)(x_2 - x_1)$

or, $y_2(s) = y_0(s)$

$+\frac{y_1(s) - y_0(s)}{h}.2h + a_2(s)2h.h$

or, $a_2(s) = \frac{\Delta y_1(s) - \Delta y_0(s)}{2!h^2} = \frac{\Delta^2 y_0(s)}{2!h^2}.$

Therefore, $A_2 = \left\{ \begin{array}{l} a_2(s) = \frac{\Delta^2 y_0(s)}{2!h^2} \\ : s \in [0,1] \end{array} \right\} = \frac{1}{2!h^2} \Delta_p^2 [y_{0L}, y_{0U}].$

In this manner,

$$A_3 = \left\{ a_3(s) = \frac{\Delta^3 y_0(s)}{3!h^3} : s \in [0,1] \right\} = \frac{1}{3!h^3} \Delta_p^3 [y_{0L}, y_{0U}], \ldots,$$

$$A_k = \left\{ a_k(s) = \frac{\Delta^k y_0(s)}{k!h^k} : s \in [0,1] \right\} = \frac{1}{k!h^k} \Delta_p^k [y_{0L}, y_{0U}].$$

Substituting these values, (4) becomes

$$\psi(x) = [y_{0L}, y_{0U}] + \frac{1}{h} \Delta_p [y_{0L}, y_{0U}](x - x_0) + \frac{1}{2!h^2} \Delta_p^2 [y_{0L}, y_{0U}](x - x_0)(x - x_1) + \ldots + \frac{1}{k!h^k} \Delta_p^k [y_{0L}, y_{0U}](x - x_0)(x - x_1)\ldots(x - x_{k-1}) \quad (7)$$

This polynomial is called as Newton's forward p-difference interpolating polynomial with interval coefficient.

3.2.2 Newton's P-difference (backward) Interpolation Formula:

In order to derive the Newton's p-difference (backward) interpolation formula, $\psi(x)$ can be taken as,

$$\psi(x) = A_0 + A_1(x - x_k) + A_2(x - x_k)(x - x_{k-1}) + \ldots + A_k(x - x_k)(x - x_{k-1})\ldots(x - x_1) \quad (8)$$

where coefficients $A_i = [a_{iL}, a_{iU}]$

$(i = 0, 1, \ldots, k)$ to be determined.

The parametric form of (3) can be written as

$$\{y_i(s) - \psi(x_i, s) = 0 : s \in [0,1], i = 0, 1, \ldots, k\} \quad (9)$$

where

$\psi(x, s) = a_0(s) + a_1(s)(x - x_k)$

$+a_2(s)(x - x_k)(x - x_{k-1})$

$+\ldots + a_k(s)(x - x_k)(x - x_{k-1})\ldots(x - x_1),$

$A_i = \left\{ \begin{array}{l} a_i(s) = a_{iL} + s(a_{iU} - a_{iL}) \\ : s \in [0,1] \end{array} \right\}$

and $y_i(s) = y_{iL} + s(y_{iU} - y_{iL})$.

Since (9) holds for all $s \in [0,1]$, $y_i(s) - \psi(x_i, s) = 0$ (10)

To determine the values of $a_i(s)$'s, substituting $x = x_k, x_{k-1}, x_{k-2}, \ldots, x_1$ in $\psi(x, s)$ and using (10) we get,

$\psi(x_k, s) = a_0(s)$ i.e., $a_0(s) = y_k(s)$.

For $x = x_{k-1}, \psi(x_{k-1}, s) = a_0(s) + a_1(s)(x_{k-1} - x_k)$,

or, $y_{k-1}(s) = y_k(s) + a_1(s)h$

or, $a_1(s) = \frac{y_k(s) - y_{k-1}(s)}{h} = \frac{\nabla y_k(s)}{h}$

Therefore, $A_1 = \left\{ a_1(s) = \frac{\nabla y_k(s)}{h} : s \in [0,1] \right\}$

$= \frac{1}{h} \nabla_p [y_{kL}, y_{kU}]$

For $x = x_{k-2}, \psi(x_{k-2}, s) = a_0(s) + a_1(s)(x_{k-2} - x_k)$
$+a_2(s)(x_{k-2} - x_k)(x_{k-2} - x_{k-1})$

$$or, y_{k-2}(s) = y_k(s) + \frac{y_k(s) - y_{k-1}(s)}{h}.2h + a_2(s)2h.h$$

$$or, a_2(s) = \frac{\nabla y_k(s) - \nabla y_{k-1}(s)}{2!h^2} = \frac{\nabla^2 y_0(s)}{2!h^2}.$$

$$\text{Therefore, } A_2 = \left\{a_2(s) = \frac{\nabla^2 y_k(s)}{2!h^2} : s \in [0,1]\right\}$$

$$= \frac{1}{2!h^2}\nabla_p^2[y_{kL}, y_{kU}].$$

In this manner,

$$A_3 = \left\{a_3(s) = \frac{\nabla^3 y_k(s)}{3!h^3} : s \in [0,1]\right\} = \frac{1}{3!h^3}\nabla_p^3[y_{kL}, y_{kU}], \ldots,$$

$$A_k = \left\{a_k(s) = \frac{\nabla^k y_k(s)}{k!h^k} : s \in [0,1]\right\} = \frac{1}{k!h^k}\nabla_p^k[y_{kL}, y_{kU}].$$

Substituting these values, (8) becomes

$$\psi(x) = [y_{kL}, y_{kU}] + \frac{1}{h}\nabla_p[y_{kL}, y_{kU}](x - x_k)$$
$$+\frac{1}{2!h^2}\nabla_p^2[y_{kL}, y_{kU}](x - x_k)(x - x_{k-1})$$
$$+\ldots+\frac{1}{k!h^k}\nabla_p^k[y_{kL}, y_{kU}](x - x_k)(x - x_{k-1})\ldots(x - x_1) \quad (11)$$

This polynomial is called as Newton's p-difference (backward) interpolating polynomial with interval coefficient.

3.3.3 Algorithm of Newton's P-difference (forward) Interpolation Formula (Pal [15]):

Step-1: Input the values of k, x^*, x_j, y_{jL} and y_{jU} for $j = 1, 2, \cdots, k$

Step-2: Calculate $y_j(p)$, h, u using following formulae

$y_j(p) = y_{jL} + p(y_{jU} - y_{jL})$ for
$j = 1, 2, \cdots, k$ and $p \in [0,1]$

$$h = x_{j+1} - x_j,\ u = \frac{x^* - x_0}{h}$$

Step-3:

Set $prod = 1$, $sum(p) = y_0(p)$;
for $j = 1$ to k do
for $i = 0$ to $k - j$ do

$dy_i(p) = y_{i+1}(p) - y_i(p)$; // $dy(p)$ represents the difference. //

$$prod = \frac{prod \times (u - j + 1)}{j};$$

$sum(p) = sum(p) + prod \times dy_0(p)$;
endfor;

Step-4: Compute

$$y_L^* = \min_{p\in[0,1]} sum(p) \text{ and } y_U^* = \max_{p\in[0,1]} sum(p);$$

Step-5: Print the values of y_L^* and y_U^* at $x = x^*$.

Note 1: Algorithm of Newton's p-difference (backward) interpolation formula can be obtained in similar manner.

4 Numerical Examples

Example 1: Find the Newton's forward and backward p-difference interpolating polynomials with interval coefficients using the following data and find $F(1.5)$ and $F(3.5)$.

x :	1	2	3	4
$F(x)$:	$[2,4]$	$[3,6]$	$[9,14]$	$[25,35]$

Solution

(i) Using Newton's p-difference formula (forward):

Since 1.5 is near to lower bound of the interval $[1,4]$, $F(1.5)$ is evaluated using forward p-difference interpolation formula.

The parametric data of the given interval-valued data is

x :	1	2	3	4
$f(x,s)$:	$2+2s$	$3+3s$	$9+5s$	$25+10s$

The difference table of above parametric data is

Table 5: Forward difference table in parametric form of Example 1.

x	$f(x,s)$	$\Delta f(x,s)$	$\Delta^2 f(x,s)$	$\Delta^3 f(x,s)$
1	$2+2s$			
		$1+s$		
2	$3+3s$		$5+s$	
		$6+2s$		$5+2s$
3	$9+5s$		$10+3s$	
		$16+5s$		
4	$25+10s$			

Here, $h=1,$ then the Newton's forward interpolation polynomial for this problem is

$$F(x)=F(x_0)+\frac{1}{h}\Delta_P F(x_0)(x-x_0)$$

$$+\frac{1}{2!h^2}\Delta_p^2 F(x_0)(x-x_0)(x-x_1)$$

$$+\frac{1}{3!h^3}\Delta_p^3 F(x_0)(x-x_0)(x-x_1)(x-x_2).$$

$$\{g(x,s):s\in[0,1]\}=\{2+2s:s\in[0,1]\}$$

$$+\{1+s:s\in[0,1]\}(x-1)$$

$$+\frac{1}{2}\{5+s:s\in[0,1]\}(x-1)(x-2)$$

$$+\frac{1}{6}\{5+2s:s\in[0,1]\}(x-1)(x-2)(x-3).$$

$$=\left\{\begin{array}{l}(2+2s)+(1+s)(x-1)+\frac{1}{2}(5+s)(x-1)(x-2)\\ +\frac{1}{6}(5+2s)(x-1)(x-2)(x-3):s\in[0,1]\end{array}\right\}$$

$$F(1.5)=\left\{\begin{array}{l}(2+2s)+(1+s)\left(\frac{1}{2}\right)\\ +\frac{1}{2}(5+s)\left(\frac{1}{2}\right)\left(-\frac{1}{2}\right)\\ +\frac{1}{6}(5+2s)\left(\frac{1}{2}\right)\left(-\frac{1}{2}\right)\left(-\frac{3}{2}\right)\\ :s\in[0,1]\end{array}\right\}$$

$$=\left\{\frac{35+40s}{16}:s\in[0,1]\right\}$$

$$=\left[\begin{array}{l}\min\left\{\frac{35+40s}{16}:s\in[0,1]\right\}\\ ,\max\left\{\frac{35+40s}{16}:s\in[0,1]\right\}\end{array}\right].$$

$$=\left[\frac{35}{16},\frac{75}{16}\right]=[2.1875,4.6875].$$

To get a visualising idea about the Newton's interval (forward) interpolating polynomial for the given data, we have plotted the curve of $g(x,s)$ considering the values of s = 0, 0.2, 0.4, 0.6, 0.8 and 1

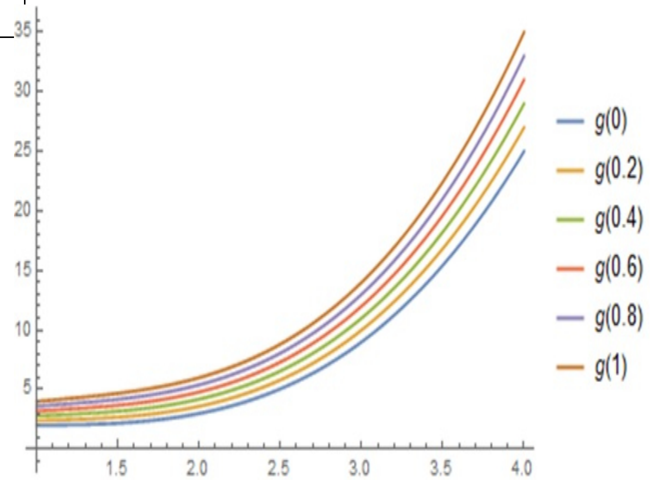

Fig. 1: Interpolating polynomial using forward difference.

(ii) Using Newton's p-difference formula (backward):

Since, 3.5 is near to 4, $F(3.5)$ is evaluated using backward p-difference interpolation formula,
The difference table of the corresponding parametric data of given interval data is given in Table 6.

Table 6: Backward difference table in parametric form of Example 1.

x	$f(x,s)$	$\nabla f(x,s)$	$\nabla^2 f(x,s)$	$\nabla^3 f(x,s)$
1	$2+2s$			
2	$3+3s$	$1+s$		
3	$9+5s$	$6+2s$	$5+s$	

4	$25+10s$	$16+5s$	$10+3s$	$5+2s$

Here, $h=1$, then the Newton’s forward interpolation polynomial for this problem is

$$F(x)=F(x_4)+\frac{1}{h}\nabla_P F(x_4)(x-x_4)$$
$$+\frac{1}{2!h^2}\nabla_P^2 F(x_4)(x-x_4)(x-x_3)$$
$$+\frac{1}{3!h^3}\nabla_P^3 F(x_4)(x-x_4)(x-x_3)(x-x_2).$$
$$=\{25+10s : s\in[0.1]\}$$
$$+\{16+5s : s\in[0,1]\}(x-4)$$
$$+\frac{1}{2}\{10+3s : s\in[0,1]\}(x-4)(x-3)$$
$$+\frac{1}{6}\{5+2s : s\in[0,1]\}(x-4)(x-3)(x-2).$$
$$=\left\{\begin{array}{l}(25+10s)+(16+5s)(x-4)\\ +\frac{1}{2}(10+3s)(x-4)(x-3)\\ +\frac{1}{6}(5+2s)(x-4)(x-3)(x-2)\\ : s\in[0,1]\end{array}\right\}$$
$$=\left\{\begin{array}{l}h(x,s)=(25+10s)+(16+5s)(x-4)\\ +\frac{1}{2}(10+3s)(x-4)(x-3)\\ +\frac{1}{6}(5+2s)(x-4)(x-3)(x-2)\\ : s\in[0,1]\end{array}\right\}$$

$$\text{And } F(3.5)=\left\{\begin{array}{l}(25+10s)+(16+5s)\left(-\frac{1}{2}\right)\\ +\frac{1}{2}(10+3s)\left(-\frac{1}{2}\right)\left(\frac{1}{2}\right)\\ +\frac{1}{6}(5+2s)\left(-\frac{1}{2}\right)\left(\frac{1}{2}\right)\left(\frac{3}{2}\right)\\ : s\in[0,1]\end{array}\right\}$$
$$=\left\{\frac{(247+112s)}{16} : s\in[0,1]\right\}$$
$$=\begin{bmatrix}\min\left\{\frac{(247+112s)}{16} : s\in[0,1]\right\}\\ ,\max\left\{\frac{(247+112s)}{16} : s\in[0,1]\right\}\end{bmatrix}$$
$$=\left[\frac{247}{16},\frac{359}{16}\right]=[15.4375, 22.4375].$$

We have plotted the curve of $h(x,s)$ considering the values of $s=0, 0.2, 0.4, 0.6, 0.8$ and 1.

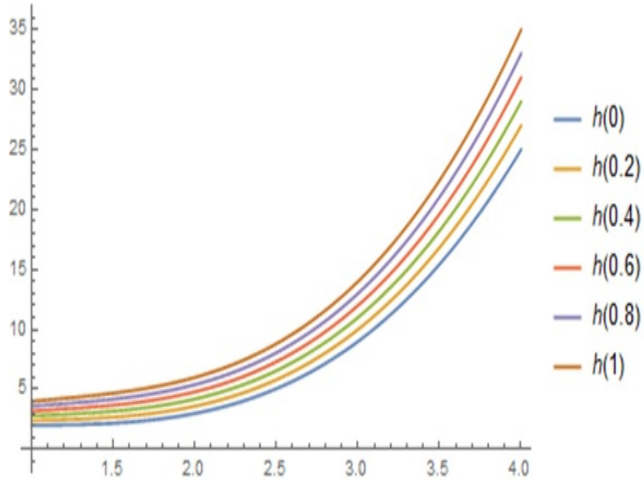

Fig. 2: Interpolating polynomial using backward difference

Example 2 The expected temperature (in the form of interval) of a city of the months November and December are given in the following

Day :	1	11	21	31	41	51	61
Temp (°C) :	[18,29]	[20,27]	[18,26]	[17,24]	[14,23]	[14,22]	[13,20]

Estimate the temperatures for the days5, 10, 30, 49and 60 using forward p-difference interpolation formula.

Solution Here the given data of the temperature $([y_{iL}, y_{iU}])$ in corresponding days (x_i) are

$$x_1=1,\quad [y_{1L}, y_{1U}]=[18,29]$$
$$x_2=11, [y_{2L}, y_{2U}]=[20,27]$$
$$x_3=21, [y_{3L}, y_{3U}]=[18,26]$$
$$x_4=31, [y_{4L}, y_{4U}]=[17,24]$$
$$x_5=41, [y_{5L}, y_{5U}]=[14,23]$$
$$x_6=51, [y_{6L}, y_{6U}]=[14,22]$$
$$x_7=61, [y_{7L}, y_{7U}]=[13,20]$$

Now we have to find the interval-valued temperature $([y_L^*, y_U^*])$ in the days $x^*=5$, 10, 30, 49 and 60.

The interpolating point are $x^* = 5$, 10, 30, 49 and 60.

Number of subintervals is $n = 6$.

The above problem has been solved using C++ language with the help of the algorithm of Newton's forward p-difference interpolation formula given in the section 4.3 in the LINUX environment and the obtained results are given in the following

x^* :	5	10	30	49	60
$[y_L^*, y_U^*]$:	[21.17,27.35]	[20.34,27.01]	[17.20,24.17]	[13.52,22.32]	[14.15,20.09]

Example 3 The following crisp data of population are taken from the corresponding interval-valued population of Example 2

Day :	1	11	21	31	41	51	61
Temp (°C):	23	25	22	20	19	16	15

Estimate the temperatures for the days 5, 10, 30, 49and 60 using forward p-difference interpolation formula.

Solution Here the given data of the population of corresponding years (x_i) are represented in the interval form $[y_{iL}, y_{iU}]$ as

$$x_1 = 1, \quad [y_{1L}, y_{1U}] = [23, 23]$$
$$x_2 = 11, [y_{2L}, y_{2U}] = [25, 25]$$
$$x_3 = 21, [y_{3L}, y_{3U}] = [22, 22]$$
$$x_4 = 31, [y_{4L}, y_{4U}] = [20, 20]$$
$$x_5 = 41, [y_{5L}, y_{5U}] = [19, 19]$$
$$x_6 = 51, [y_{6L}, y_{6U}] = [16, 16]$$
$$x_7 = 61, [y_{7L}, y_{7U}] = [15, 15]$$

Now we have to find the temperatures y^* on the days $x^* = 5$, 10, 30, 49 and 60.

The interpolating points are $x^* = 5$, 10, 30, 49 and 60.

Number of subintervals is $n = 6$.

The above problem has been solved using C++ language with the help of the algorithm of Newton's forward p-difference interpolation formula and existing interpolation formula given in the Section 4.3 in the LINUX environment and the obtained results are given in the following

	x^*	5	10	30	49	60
P-diff. formula	$[y_L^*, y_U^*]$	[24.96,24.96]	[25.17,25.17]	[20.11,20.11]	[16.78,16.78]	[14.50,14.50]
Existing formula	y^*	24.96	25.17	20.11	16.78	14.50

The graph of the interpolating polynomials using known data and obtained data of the Example 2 and Example 3 has been plotted in the Fig. 3.

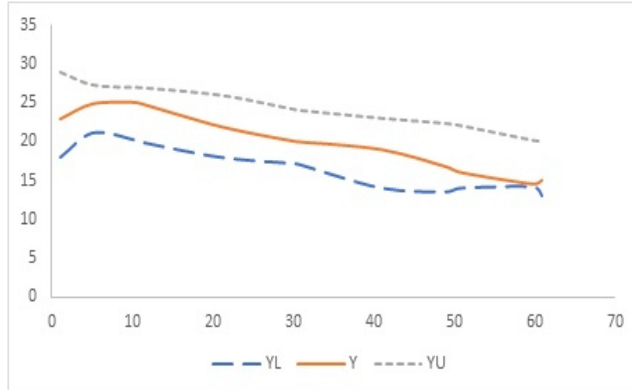

Fig.3: Interpolating polynomial of Example 2 and Example 3.

5 Discussions

Analysing the definition of p-differences, p-difference interpolation formulae, numerical examples 1-3 and the figures 1-3 we can note the followings observations:

- If both the bounds of the interval-valued data points are equal i.e., the data points are real numbers, then the p-differences and corresponding p-difference interpolation formulae become the traditional interpolation formulae in crisp environment.
- From Fig. 1 and Fig. 2, it is clear that the Newton backward p-difference interval interpolating polynomial and Newton forward p-difference interval interpolating polynomial are identical.
- The interpolating results of example 2 and 3 give that the functional values of any interpolating point always lie between the bounds of corresponding interval interpolating result and the geometrical justification is shown in Fig. 3 which is discussed in next bullet.
- From Fig. 3, it is observed that the graph of interpolating polynomial (Y) using crisp data given in Example 3 always lies between the bounds (YL and YU) of the graph of interval interpolating polynomial using the data of Example 2.

6 Conclusion

In this work, using parametric representation of interval, the definitions of finite p-differences (forward/backward) have been introduced. Then with the help of these p-differences, Newton's interpolation formulae for interval-valued functions using finite p-differences (forward/backward) have been derived. Analysing numerical examples, it has been shown that the results obtained from both Newton's p-difference interpolation formulae (forward/backward) be the same. Finally, it should be concluded that the proposed Newton's p-difference interpolation formulae (forward/backward) be the generalization of the traditional Newton's interpolation formulae (forward/backward).

Acknowledgement: The second author acknowledges the support of Department of Science and Technology, Ministry of Science & Technology, Government of India under INSPIRE Fellowship (vide sanction order number: DST/INSPIRE Fellowship/2018/IF180100) for accomplishing this research work.

References

[1] H. Akima, A new method of interpolation and smooth curve fitting based on local procedures, Journal of the ACM (JACM)., **17(4)**, 589-602, 1970.

[2] D. Kincaid and W. Cheney, Numerical Analysis, Brooks/Cole Publishing Company, Pacific Grove, California, USA, 2nd edition, 1996.

[3] G.F. Gerald and P.O. Wheatly, Applied Numerical Analysis, 5th edition, Addision-wesley, Reading, Mass., 1994.

[4] M. Barrault, Y. Maday, N.C. Nguyen, and A.T. Patera, An 'empirical interpolation 'method: application to efficient reduced-basis discretization of partial differential equations. Comptes Rendus Mathematique., **339(9)**, 667-672, 2004.

[5] Z. Huang and Q. Shen, Fuzzy interpolation and extrapolation: A practical approach, IEEE Transactions on Fuzzy Systems., **16(1)**, 13-28, 2008.

[6] R. E. Moore, Methods and applications of interval analysis. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1979.

[7] R. E. Moore, R. B. Kearfott and M.J. Cloud, Introduction to interval analysis, Siam., Philadelphia, PA, USA, 2009.

[8] M. Ramezanadeh, M. Heidari, O.S. Fard, and A. H. Borzabadi, On the interval differential equation: novel solution methodology, Advances in Difference Equations., 2015 **(1)**, 338, 2015.

[9] S.M. Markov, some interpolation problems involving interval data, Interval Computations., **3**, 164-182, 1993.

[10] A.T. Ali, M. M Khater, R. A. Attia, A. H. Abdel-Aty, and D. Lu, Abundant numerical and analytical solutions of the generalized formula of Hirota-Satsuma coupled KdV system. Chaos, Solitons & Fractals, 109473, 2019.

[11] S. Owyed, M. A. Abdou, A. H. Abdel-Aty, S. S Ray, New optical soliton solutions of nolinear evolution equation describing nonlinear dispersion. Communications in Theoretical Physics., **71(9)**, 1063, 2019.

[12] S. Owyed, M. A. Abdou, A. H. Abdel-Aty, W. Alharbi, and R. Nekhili, Numerical and approximate solutions for coupled time fractional nonlinear evolutions equations via reduced differential transform method. Chaos, Solitons & Fractals, 109474, 2019.

[13] A. E. T. Elgendy, A. H. Abdel-Aty, A. A. Youssef, M.A. Khder, K. Lotfy and S. Owyed, Exact solution of Arrhenius equation for non-isothermal kinetics at constant heating rate and n-th order of reaction. Journal of Mathematical Chemistry., 1-17, 2020.

[14] G.M. Ismail, H. R. Abdl-Rahim, A. Abdel-Aty, R. Kharabsheh, W. Alharbi and M. Abdel-Aty, An analytical solution for fractional oscillator in a resisting medium. Chaos, Solitons & Fractals., **130**, 109395, 2020.

[15] M. Pal, Numerical analysis for scientists and engineers, theory and C programs, Alpha Science International Limited, the University of California, USA, 2007.

Md Sadikur Rahman completed his M.sc degree in the year of 2017 from the university of Kalyani, west Bengal, India. Presently, he is working as a junior research fellow at the department of Mathematics, University of Burdwan, West Bengal, India.

His interest of research area is non-linear optimization in interval environment and its application in inventory.

Md Akhtar is a Ph.D. Scholar in the Department of Mathematics at The University of Burdwan, India. He obtained his M. Sc. degree in Mathematics (Applied Stream) from The University of Burdwan, India. He has received gold medal for being first in the first class in the M.Sc. Examination, 2016 in Mathematics (Applied Stream).

His research interests include Soft Computing, Interval Mathematics, Interval ranking, etc.

Ali Akbar Shaikh is currently working as an Assistant Professor in Mathematics at The University of Burdwan, Burdwan, West Bengal, India. He was a postdoctoral fellow at School of Engineering and Sciences of Tecnológico de Monterrey, México. He has obtained the award SNI of level 1 (out of 0-3) given by National System of Researchers of México from the government of México in the year 2017.

He obtained MSc in Applied Mathematics from University of Kalyani, India, MPhil and PhD in Mathematics from the University of Burdwan, India. Dr. Shaikh has published 44 research papers in different peer-reviewed and JCR international journals. His interest of research area is non-linear optimization in interval environment and its application in inventory.

Asoke Kumar Bhunia is a Professor at the Department of Mathematics, The University of Burdwan, West Bengal, India. He obtained his PhD in Mathematics and MSc in Applied Mathematics from Vidyasagar University, India. His research interests include computational optimisation, soft computing, interval mathematics, and interval ranking.

He has published over 125 research papers in various national and international journals of repute.He is a reviewer of several SCI journals. He has guided 11 PhD and two MPhil students. He is an author of four research monographs and six book chapters. He is an INSA Visiting Fellow and former Associate Editor of the Springer's journal, *OPSEARCH*.