March 2021

NODE-INDEPENDENT METHOD FOR GASTROENTEROLOGICAL SIGNAL PROCESSING BASED ON CUBIC SPLINES

S.A. Bakhromov

“Bulletin of TUIT: Management and Communication Technologies”, ab.shaxnoza84@gmail.com

Follow this and additional works at: https://uzjournals.edu.uz/tuitmct

Part of the Data Science Commons

Recommended Citation
Available at: https://uzjournals.edu.uz/tuitmct/vol4/iss1/5

This Article is brought to you for free and open access by 2030 Uzbekistan Research Online. It has been accepted for inclusion in Bulletin of TUIT: Management and Communication Technologies by an authorized editor of 2030 Uzbekistan Research Online. For more information, please contact sh.erknov@edu.uz.
**NODE-INDEPENDENT METHOD FOR GASTROENTEROLOGICAL SIGNAL PROCESSING BASED ON CUBIC SPLINES**

H.N. Zaynidinov, S.A. Bakhromov, B.R. Azimov

**Abstract.** This paper discusses a local cubic spline function built independently of node points using basic functions. The size of the calculations required to find the parameters to be determined during the construction of the spline function does not depend on the number of node points. Local-based splines are used to build such spline functions. Restoration of the gastroenterological signal was performed on the basis of the spline-function model discussed in the article. The result of a cubic spline-function error independent of the node points was compared with the result of the Lagrange classical polynomial error (Table 2).

**Keywords.** local basis functions, local cubic spline, Lagrange classical polynomial, gastroenterological signal, interpolation.

**Introduction**

Today, spline-function models are widely used by researchers in the field of digital signal processing [1, 2]. Because the accuracy of spline functions is higher than that of classical polynomials, and the algorithms based on them require less computation. Existing classical interpolation models, their application in signal recovery and digital processing algorithms are performed depending on the node points. In this case, a large number of node points are required for these models to have a high degree of convergence. In cases where the nodes are point-dependent, the order of the system of equations formed in the construction of the model under consideration increases. And this complicates the process of solving a system of equations and does not ensure a high level of accuracy of the model [3, 4, 14-20].

The process of digital processing of signals of the spline model discussed in this article was carried out independently of the node points. In this case, the increase in the degree of convergence of the model under consideration does not depend on the increase in the order of the system of equations. Methods for determining nonlinear spline functions, their derivatives, and estimating errors are performed in the same way as for simple interpolation splines [5, 9,10].

1. **Local base functions**

Let us be given the following function [12].

\[ G(x,t) = (x-t)^3, \quad x \geq t \]
\[ = 0, \quad x < t \]  

(1)

Assume that the \( \{x_i\} \) node points on the \( OX \) axis are defined in steps \( h \) as follows.

\[ x_{i+1} = x_i + h, \quad i = 1,2,\ldots,M, \quad G(x,t) \]

we enter the fourth-order divisor of the function separately for the \( x_{i-2}, x_i-1, x_i, x_i+1, x_i+2 \) node points on the variable \( i \):

\[ \phi_i(x) = G(x_{i-2} \cdot x_i-1 \cdot x_i \cdot x_i+1 \cdot x_i+2), \]
\[ i = 3,4,\ldots,M-2 \]

The fourth-order difference of the \( \phi(x) \) function is determined by the following formula:

\[ f(x_{i-2} \cdot x_i-1 \cdot x_i \cdot x_i+1 \cdot x_i+2) = \]
\[ = f(x_{i-1} \cdot x_i \cdot x_i+1 \cdot x_i+2) - f(x_{i-2} \cdot x_i-1 \cdot x_i \cdot x_i+1) \]
\[ \cdot \]
\[ = x_{i-2} - x_i+2 \]

In turn, \( x_{i-1} \cdot x_i \cdot x_i+1 \cdot x_i+2 \) and \( x_{i-2} \cdot x_i-1 \cdot x_i \cdot x_i+1 \) are also calculated sequentially as above.

After some simplification, the calculation formula for the fourth-order subtraction difference of the \( G(x,t) \) function will look like this:

\[ \phi_i(x) = G(x_{i-2} \cdot x_i-1 \cdot x_i \cdot x_i+1 \cdot x_i+2) = \]
\[ = \frac{1}{4h}((x-x_i+2)^3 - 4(x-x_i+1)^3) + 6(x-x_i)^3 - 4(x-x_i-1)^3 + (x-x_i-2)^3 \]

(2)

Based on the features discussed above, the following forms the basis in the space of tertiary splines
\[ S_i(x) = \frac{\varphi_i(x)}{\varphi_i(x_i)}, \quad i = 3, 4, 5, \ldots, M - 2 \] (3)

Let's look at the features.
This function forms the basis in the space of cubic splines and has the following features:

1) Smoothness
\[ S_i(x) \in C^2[x_i, x_M] \] (31)

2) Locality
\[ S_i(x) > 0, \quad x \in (x_i - 2, x_i + 2) \]
\[ S_i(x) = 0, \quad x \not\in (x_i - 2, x_i + 2), \quad i = 3, 4, 5, \ldots, M - 2 \] (32)

Here we see as proof
\[ \varphi_i(x) = 0 \quad \text{if} \quad x \not\in (x_i - 2, x_i + 2) \]

Now let's calculate \( \varphi_i(x_{i-2}), \varphi_i(x_{i-1}), \varphi_i(x_i), \varphi_i(x_{i+1}) \) and \( \varphi_i(x_{i+2}) \):

\[ \varphi_i(x_{i-2}) = \frac{1}{4h_i^3} \left( (x_i - 2 - x_i + 2)^3 + \right. \]
\[ - 4(x_i - 2 - x_{i+1})^3 + 6(x_i - 2 - x_i)^3 \]
\[ - 4(x_i - 2 - x_{i-1})^3 + (x_i - 2 - x_i^2)^3 \left. \right) = \]
\[ = \left\{ G(x,t) = (x-t)^3 \right\} = \left\{ \begin{array}{ll}
(x-t)^3, & x \geq t \\
0, & x > t
\end{array} \right. = 0 \]
\[ \varphi_i(x_{i-2}) = (4h_i^3 - 4(3h_i^3 - 6(2h_i^3 - 4h_i^3 =
\left[ 64h_i^3 - 108h_i^3 + 48h_i^3 - 4h_i^3 \right] = 0 \]
\[ \varphi_i(x_{i-2}) = 0 \] (4)

The rest according to (32) are also found as follows

\[ \varphi_i(x_{i-1}) = \frac{1}{24h} \] (5)
\[ \varphi_i(x_i) = \frac{1}{6h} \] (6)
\[ \varphi_i(x_{i+1}) = \frac{1}{24h} \] (7)

As above, we find
\( \varphi_i(x_{i-2}), \varphi_i(x_{i-1}), \varphi_i(x_i), \varphi_i(x_{i+1}), \varphi_i(x_{i+2}) \)

based on the following nodes.

That is:
\[ \varphi_i(x_{i-2}) = G(x, x_i - 4, x_{i-2}, x_{i-3}, x_{i-1}, x_i) =
\]
\[ = \frac{1}{4h^4} \left[ (x-x_i)^3 - 4(x-x_{i-1})^3 \right] \]
\[ + 6(x-x_{i-2})^3 - 4(x-x_{i-3})^3 + (x-x_i)^3 \right] =
\]
\[ = \frac{1}{4h^4} \left[ C_0(x-x_i)^3 - C_1(x-x_{i-1})^3 \right. \]
\[ + C_2(x-x_{i-2})^3 - C_3(x-x_{i-3})^3 \]
\[ + C_4^2(x-x_{i-4})^3 \] =
\[ = \frac{1}{4h^4} \sum_{j=-4}^{0} (-1)^{j+4} + j C_{4+j}(x-x_i)^3 \]
\[ = \frac{1}{4h^4} \sum_{j=-4}^{0} (-1)^{j+4} + j \] (4)

We perform the replacement as follows, which will be
\[ x_{i+j} = h(t-j) \]

In that case
\[
\varphi_{i-2}(x) = \frac{1}{4h^4} \sum_{j=-4}^{0} (-1)^{4-j} j \varphi_{i+j}(x) + \frac{1}{h^3} \left( \frac{x-x_i + x_i - x_{i-4} - j}{h} \right) + \frac{1}{24h} \left( -t^3 + 4t^2 - 6t + 3 \right)
\]

\[
\varphi_{i+2}(x) = \frac{1}{24h^3} \left( \frac{x-x_i - h}{h} \right)^3 = \frac{1}{24h} \left( \frac{x-x_i}{h} \right)^3
\]

Based on the above, we also substitute the \( \varphi_{i-1}(x), \varphi_i(x), \varphi_{i+1}(x) \) for the variable \( t \) to form the following equation:

\[
\varphi_{i-2}(x) = 0 \\
\varphi_{i-1}(x) = \frac{1}{24h}(1-t)^3 \\
\varphi_i(x) = \frac{1}{24h}(3t^3 - 6t^2 + 4) \\
\varphi_{i+1}(x) = \frac{1}{24h}(1 + 3t + 3t^2 - 3t^3) \\
\varphi_{i+2}(x) = \frac{1}{24h}(t^3)
\]

We create the basis functions based on (4) - (8) and (9) - (13).

\[
S_{i-2}(x) = \frac{\varphi_{i-2}(x)}{\varphi_{i-2}(x - i)} = 0
\]

\[
S_{i-1}(x) = \frac{\varphi_{i-1}(x)}{\varphi_{i-1}(x - i)} = \frac{1}{24h}(1-t)^3
\]

\[
S_i(x) = \frac{\varphi_i(x)}{\varphi_i(x - i)} = \frac{1}{24h}(3t^3 - 6t^2 + 4)
\]

\[
S_{i+1}(x) = \frac{\varphi_{i+1}(x)}{\varphi_{i+1}(x + i)} = \frac{1}{24h}(1 + 3t + 3t^2 - 3t^3)
\]

\[
S_{i+2}(x) = \frac{1}{24h}(t^3)
\]

We construct the basic functions based on (4) - (8) and (9) - (13).

\[
\varphi_{i-2}(x) = \frac{1}{4h^4} \sum_{j=-4}^{0} (-1)^{4-j} j \varphi_{i+j}(x) + \frac{1}{h^3} \left( \frac{x-x_i + x_i - x_{i-4} - j}{h} \right) + \frac{1}{24h} \left( -t^3 + 4t^2 - 6t + 3 \right)
\]

Now let's do a switch at \( x \in [x_i, x_{i+1}], \) it will be \( x-x_{i+j} = h(t-j) \).

In that case
\[ S_{i+2}(x) = \frac{\varphi_{i+2}(x)}{3} \quad \frac{1}{4} \quad \frac{1}{6h} \quad = \frac{1}{4} \quad = \frac{1}{6h} \quad = \frac{1}{4} \quad (18) \]

The values of the \( f(x) \) function at the
\( f_i = f(x_i), \quad i = 3, M - 2 \) node points are given
\[ x_4 = a + \frac{x M - 3}{b} \]

Consider the following function.
\[ S(x) = \sum_{i=3}^{M-2} f_i S_i(x), \quad x \in (x_4, x M - 3) \]

From this function
\[ S_i(x) > 0, \quad x \in (x_4 - x_i+2) \]
\[ S_i(x) = 0, \quad x \not\in (x_4 - x_i+2) \]

the local condition is required.

In that case, the values at the node point of
the \( S_i(x) \) function based on the localization
condition are as follows.
\[ S(x_i) = \sum_{j=1}^{i+1} f_j S_j(x_i), \quad i = 4, M - 3 \]

For simplicity
\[ S(x_i) = \sum_{p=-1}^{i} f_i p S_i + p (x_i) \]

That is
\[ S(x_i) = f_i - 1 S_i - 1 (x_i) + f_i S_i (x_i) + \]
\[ + f_i + 1 S_i + 1 (x_i), \quad i = 4, M - 3 \]

The \( S_i(x) \) function at values p = 0.1 is as follows
\[ S_i + p (x_i) = S_i - p (x_i) = S_i (x_i + p) = \]
\[ = S_i (x_i - p) = a p \]

has properties
\[ a p = \begin{cases} 1 & \text{aep} \quad p = 0 \\ 0.25 & \text{aep} \quad p = 1 \end{cases} \]

That is, at p = 0
\[ S_i (x_i) = S_i (x_i) = S_i (x_i) = S_i (x_i) = 1 \]

really
\[ S_i (x) = \frac{\varphi_i (x)}{\varphi_i (x_i)} , \]

in this
\[ S_i (x_i) = \frac{\varphi_i (x_i)}{\varphi_i (x_i)} = 1, \]

The case p = 1 is similar.
As a result
\[ S(x_i) = 0.25 f_i - 1 + f_i + 0.25 f_i + 1 = \]
\[ = f + 0.25 (f_i - 1 + f_i + 1), \quad i = 4, M - 3 \]

We now express the \( f_i - 1, f_i + 1 \) through the
\( f_i \), for which we spread them to the Taylor series
\[ f_i - 1 = f_i + h f_i + \frac{h^2}{2} f_i ' ' \]
\[ f_i + 1 = f_i - h f_i + \frac{h^2}{2} f_i ' ' \]

By adding these expressions we get the following
\[ f_i - 1 + f_i + 1 = 2 f_i + \frac{h^2}{2} (f_i ' ' + f_i ' ' ) \]
\[ f_i - 1 + f_i + 1 = 2 f_i + O(h^2) \]
\[ 0.25 (f_i - 1 + f_i + 1) = 0.5 f_i + O(h^2) \]

In that case
\[ S(x_i) = f_i + (- f_i - 1 + f_i + 1) = f_i + \frac{1}{2} \]

From this
\[ S(x_i) = \frac{3}{2} f_i + O(h^2) \]

We will have \( S(x_i) = \frac{3}{2} f_i + O(h^2) \).

The smoother the \( f(x) \) function, the closer
it is to \( f_i \) in \( (f_i - 1 + f_i + 1) / 2 \) calculations.

And the \( S_i(x) \) spline function is close to the
\( k \frac{3}{2} f(x) \) at the node points then we find the
next approximate function
\[ S^* (x_i) = \frac{2}{3} S(x_i) = \frac{2}{3} \frac{3}{2} f_i + O(h^2) \]
\[ S^* (x_i) = f_i + O(h^2) \]

in which case it can be obtained
\[ S^* (x) = \frac{2}{3} M - 2 \sum_{i=3}^{M} f_i S_i(x) \]
\[ S^*(x) = \frac{2}{3} S_{i-1}(x) f_{i-1} + \frac{2}{3} S_i(x) f_i + \frac{2}{3} S_{i+1}(x) f_{i+1} + \frac{2}{3} S_{i+2}(x) f_{i+2} \]

We enter the following notation, where
\[ t = \frac{x-x_i}{h}, x = x_i + th \]

\[ \psi_1(t) = \frac{2}{3} S_{i-1}(x); \quad \psi_2(t) = \frac{2}{3} S_i(x); \]
\[ \psi_3(t) = \frac{2}{3} S_{i+1}(x); \quad \psi_4(t) = \frac{2}{3} S_{i+2}(x) \]

In that case
\[ S^*(x) = \psi_1(t) f_{i-1} + \psi_2(t) f_i + \psi_3(t) f_{i+1} + \psi_4(t) f_{i+2} \]

It is not difficult to calculate \( \psi_i(t) \).

We will see one of them count
\[ \psi_1(t) = \frac{2}{3} S_{i-1}(x) = \frac{2}{3} (1-t)^3 = \frac{1}{6} (1-t)^3 \]
the rest are calculated similarly.
\[ \psi_2(t) = \frac{1}{6} (3t^3 - 6t^2 + 4) \]
\[ \psi_3(t) = \frac{1}{6} (1 + 3t + 3t^2 + 3t^3) \]
\[ \psi_4(t) = \frac{1}{6} t^3 \]

As a result, we write the general view of a tertiary spline function that is independent of node points as follows
\[ S^*(x) = \sum_{j=1}^{4} \psi_j(t) f_{i-2+j} \]

Here
\[
\begin{align*}
\psi_1(t) &= \frac{1}{6} (1-t)^3 \\
\psi_2(t) &= \frac{1}{6} (3t^3 - 6t^2 + 4) \\
\psi_3(t) &= \frac{1}{6} (1 + 3t + 3t^2 + 3t^3) \\
\psi_4(t) &= \frac{1}{6} t^3 
\end{align*}
\]

2. Digital processing of gastroenterological signals based on developed algorithms.

The construction of cubic spline models was performed on the basis of the gastroenterological signal given in Table 1 as the initial data [3, 6, 7]. Based on the above sequence, a cubic spline construction program in the MATLAB software environment of the gastroenterological signal given in Table 1 was developed and used in the processing of the gastroenterological signal [8, 11, 13]. The algorithm of this program is shown in Figure 2.

<table>
<thead>
<tr>
<th>№</th>
<th>X[i]</th>
<th>Y[i]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>6,370,065</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>6,370,065</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>5,835,327</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>5,300,590</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>4,765,852</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>5,567,959</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>6,637,434</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>6,904,803</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>6,904,803</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>6,370,065</td>
</tr>
<tr>
<td>11</td>
<td>11</td>
<td>6,370,065</td>
</tr>
<tr>
<td>12</td>
<td>12</td>
<td>5,835,327</td>
</tr>
<tr>
<td>13</td>
<td>13</td>
<td>5,300,590</td>
</tr>
<tr>
<td>14</td>
<td>14</td>
<td>5,033,221</td>
</tr>
<tr>
<td>15</td>
<td>15</td>
<td>5,033,221</td>
</tr>
<tr>
<td>16</td>
<td>16</td>
<td>5,300,590</td>
</tr>
<tr>
<td>17</td>
<td>17</td>
<td>6,102,696</td>
</tr>
<tr>
<td>18</td>
<td>18</td>
<td>6,904,803</td>
</tr>
<tr>
<td>19</td>
<td>19</td>
<td>7,172,172</td>
</tr>
<tr>
<td>20</td>
<td>20</td>
<td>7,172,172</td>
</tr>
<tr>
<td>21</td>
<td>21</td>
<td>6,637,434</td>
</tr>
<tr>
<td>22</td>
<td>22</td>
<td>6,102,696</td>
</tr>
<tr>
<td>23</td>
<td>23</td>
<td>5,567,959</td>
</tr>
<tr>
<td>24</td>
<td>24</td>
<td>5,033,221</td>
</tr>
<tr>
<td>25</td>
<td>25</td>
<td>4,498,483</td>
</tr>
<tr>
<td>26</td>
<td>26</td>
<td>3,963,745</td>
</tr>
<tr>
<td>27</td>
<td>27</td>
<td>3,696,376</td>
</tr>
<tr>
<td>28</td>
<td>28</td>
<td>3,963,745</td>
</tr>
<tr>
<td>29</td>
<td>29</td>
<td>3,963,745</td>
</tr>
<tr>
<td>30</td>
<td>30</td>
<td>4,498,483</td>
</tr>
<tr>
<td>31</td>
<td>31</td>
<td>5,033,221</td>
</tr>
<tr>
<td>32</td>
<td>32</td>
<td>5,835,327</td>
</tr>
<tr>
<td>33</td>
<td>33</td>
<td>6,102,696</td>
</tr>
<tr>
<td>34</td>
<td>34</td>
<td>6,637,434</td>
</tr>
<tr>
<td>35</td>
<td>35</td>
<td>6,904,803</td>
</tr>
<tr>
<td>36</td>
<td>36</td>
<td>7,172,172</td>
</tr>
<tr>
<td>37</td>
<td>37</td>
<td>7,172,172</td>
</tr>
<tr>
<td>38</td>
<td>38</td>
<td>7,172,172</td>
</tr>
<tr>
<td>39</td>
<td>39</td>
<td>7,439,541</td>
</tr>
<tr>
<td>40</td>
<td>40</td>
<td>6,904,803</td>
</tr>
<tr>
<td>41</td>
<td>41</td>
<td>6,370,065</td>
</tr>
<tr>
<td>42</td>
<td>42</td>
<td>5,835,327</td>
</tr>
</tbody>
</table>
Fig. 1. Results of gastroenterological signal recovery.

Table 2. Error results in the process of digital processing of the gastroenterological signal.

<table>
<thead>
<tr>
<th>№</th>
<th>Model types</th>
<th>Absolute error</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Lagrange classical interpolation model</td>
<td>1.0493</td>
</tr>
<tr>
<td>2.</td>
<td>A cubic spline model that does not depend on node points</td>
<td>0.1025</td>
</tr>
</tbody>
</table>
3. Conclusion

This paper discusses a local cubic spline function built independently of node points using basic functions. And was used in the digital processing of the gastroenterological signal. The process of gastroenterological signal recovery was performed using the proposed method. The result showed that the method of tertiary spline functions independent of node points showed high accuracy in digital processing of signals (Figure 1).

The proposed cubic spline model has the following features compared to classical interpolation models:

- good proximity to the object when interpolating gastroenterological signals;
- the construction of the model is independent of node points and is much simpler than the classical polynomials;

![Algorithm of cubic spline construction program.](image)
that the algorithm for determining the parameters of the spline is simple and convenient.

Hence, the use of cubic spline models built independently of the node points according to Table 2 gives good results in the digital processing of gastroenterological signals.

References


