UNITECH 2023 GABROVO

INTERNATIONAL SCIENTIFIC CONFERENCE
17-18 November 2023, GABROVO

DISCRETE DATA APPROXIMATIONS AND THEIR APPLICATION IN ENGINEERING RESEARCH

Milena Racheva

Technical University of Gabrovo

Abstract

In this paper we present an important aspect in engineering practice. Namely: how to use effectively the data obtained as a result of measurements and approximations of required dependences and parameters. Our aim is to show different approaches to such computations and their philosophy for the engineering applications.

Emphasis is placed on the approximation of a limited discrete set of data in order to get Fourier harmonics, when the periodic function is unknown. This idea is supported by realistic and numerically solved examples.

Keywords: Fourier coefficient, interpolation, metric approximation

1. INTRODUCTION AND MOTIVATION

Very often in engineering applications, a process (function) is represented in a discrete form, i.e. it is tabulated as a result of measurement:

t	t_1	t_2	$\dots$	t_n
$f(t)$	$f(t_1)$	$f(t_2)$	$\dots$	$f(t_n)$

Table 1

Then $(t_i, f(t_i))$ are given points (nodes), but (usually) *f(t)* is unknown. Let *g(t)* be an approximating function for *f(t)* and *f(t) ~ g(t)*. The approximation of the data is determined by the position of the function *g(t)* to the given points.

There are two approaches for data approximation – interpolation and metric [1].

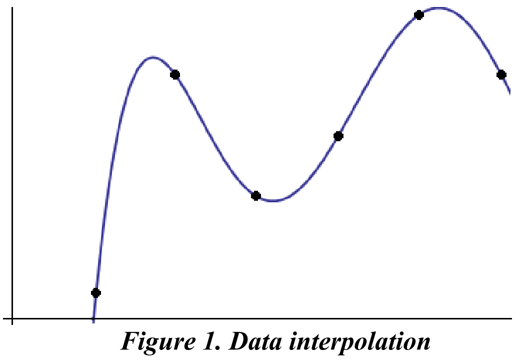

Figure 1. Data interpolation

The first one is when the approximating function passes through the given points (Fig. 1).

However, very often data are obtained after measurements and the errors are inevitable. So, once we have defined a "distance" between a point and a curve/surface, metrical method for approximation could be used, which requires that the sum of the distances between the approximate function and the data be minimal (Fig. 2). The most popular among these methods is called least squares method/regression [2].

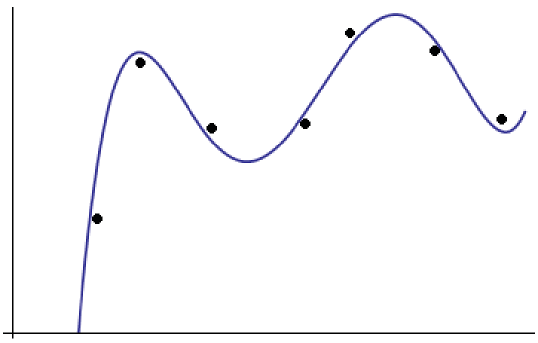

Figure 2. Metric approximation

Depending on the nature of the engineering research, an interpolation or a metric method of approximation is resorted to. We will illustrate this philosophy by examples from electrical engineering practice for purpose to obtain harmonics on the base of discrete measurement data.

2. APPLICATION

We will restrict ourselves to the case when $f(t)$ is a 2π-periodic function.

Then the k-th harmonic is [3]:

$$a_k \cos kt + b_k \sin kt,$$

where a_k and b_k are the k-th Fourier coefficients. These coefficients could be determined by numerical integration as well as least squares method could be implemented. Note that all are designed for data that is in tabular form.

Consider the following data:

i	t_i	$f(t_i)$
0	0	0.25
1	$\frac{\pi}{6}$	1.9
2	$\frac{\pi}{3}$	6.1
3	$\frac{\pi}{2}$	11.8
4	$\frac{2\pi}{3}$	17.9
5	$\frac{5\pi}{6}$	15
6	π	14.2
7	$\frac{7\pi}{6}$	11.1
8	$\frac{4\pi}{3}$	7
9	$\frac{3\pi}{2}$	4.2
10	$\frac{5\pi}{3}$	3
11	$\frac{11\pi}{6}$	1.9
12	2π	0.19

Table 2. Tabular data for f(t)

We will find the Fourier expansion of the function $f(t)$ (unknown!):

$$\frac{a_0}{2} + \sum_{k=1}^{\infty} \left(a_k \cos kt + b_k \sin kt \right),$$

where

$$a_k = \frac{1}{\pi} \int_0^{2\pi} f(t) \cos kt \, dt, \; k = 0,1,2,...$$

$$b_k = \frac{1}{\pi} \int_0^{2\pi} f(t) \sin kt \, dt, \; k = 1,2,3,... \qquad (1)$$

Obviously, the integrals from *(1)* have to be calculated numerically or the data for $f(t)$ have to be approximated.

Using these two approaches, we will find a trigonometric polynomial of order three. Wolfram Cloud resources [4] were used for the calculations. Therefore, some of the notations in the present work are consistent with Wolfram Language [5].

First we will apply interpolation approach, using numerical integration. Let us recall that the Newton-Cotes formulas are based on the strategy of piecewise interpolation of the functions under the integral sign.

2.1. Rectangular rule

We employ a series of zero-order polynomials, which are constants, and mesh parameter is $\frac{\pi}{6}$. Then we obtain the following approximations of a_k and b_k:

$$\tilde{a}_k = \frac{2}{m} \sum_{i=0}^{m-1} f(t_i) \cos kt_i,$$

$$\tilde{b}_k = \frac{2}{m} \sum_{i=0}^{m-1} f(t_i) \sin kt_i,$$

and here $m = 12$.

Thus the third-order trigonometric polynomial is:

$$T_3^{Rect}(t)=7.86-6.8654\cos t+3.6124\sin t-0.605\cos 2t-1.68875\sin 2t+0.308333\cos 3t-0.616667\sin 3t. \quad (2)$$

2.2. Trapezoidal rule

The functions under integration from *(1)* are interpolated by means of piecewise linear polynomials and then

$$\tilde{a}_k=\frac{2}{m}\left\{\frac{f(t_0)+f(t_m)}{2}+\sum_{i=1}^{m-1} f(t_i)\cos kt_i\right\},$$

$$\tilde{b}_k=\frac{2}{m}\left\{\frac{f(t_0)+f(t_m)}{2}+\sum_{i=1}^{m-1} f(t_i)\sin kt_i\right\},$$

and here $m = 12$.

Then we obtain the following third-order trigonometric polynomial:

$$T_3^{Trap}(t)=7.86-6.8654\cos t+3.6124\sin t-0.605\cos 2t-1.6887\sin 2t+0.3033\cos 3t-0.628\sin 3t. \quad (3)$$

2.3. Metric approach

In line with the idea to overcome effects of measurement errors, we can approximate the data for $f(t)$ by appropriate polynomials (functions) and after that to get Fourier coefficients for the function which approximates $f(t)$ [1,3].

As can be seen from Fig. 3, the data we are working with does not follow a linear dependence at all.

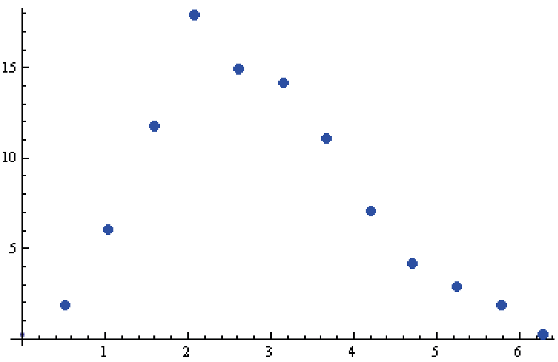

Figure 3. Data from Table 2

It seems appropriate to look for an approximating function by least squares method in the form of a second or third degree polynomial. However, after finding these approximations and calculating relevant indicators (see Table 3), we stop at and choose an approximation with a polynomial of degree four.

Polynomial Degree	AIC	RSquared	Standard Deviation
Second	72.32	0.765	10.19
Third	65.94	0.880	7.2800
Fourth	61.96	0.927	5.6670

Table 3. Least squares method indicators for 2nd, 3rd and 4th-degree polynomial approximation of f(t)

The fourth-order approximate function is Fig. 4):

$$P_4(t)=-1.01847+7.77005t+2.01773t^2-1.36192t^3.+0.135941t^4.$$

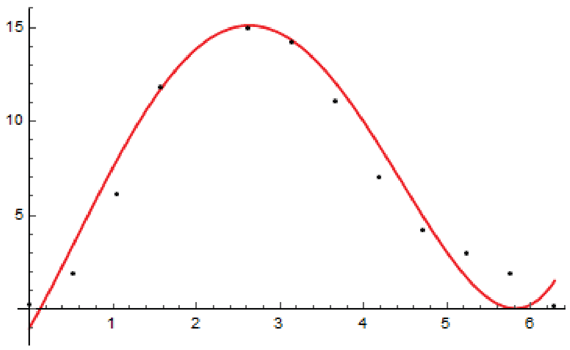

Figure 4. The 4th-order approximating polynomial

Substituting into the integrals from *(1)* $P_4(t)$ for $f(t)$, we get the following trigonometric polynomial:

$$\hat{T}_3(t)=7.86225-6.8634\cos t+3.353\sin t-0.49239\cos 2t+0.11785\sin 2t-0.11814\cos 3t-0.11386\sin 3t. \quad (4)$$

2.4. Piecewise metric approach

Taking into account the manner which the given points (Table 2) are situated in, we decided to propose a new idea for piecewise implementation of least squares method.

Let us split the tabulated data into two parts and in both of them to employ linear least square method (see Fig. 5).

We obtain the following piecewise approximation of *f(t)*:

$$P_1(t)=\begin{cases} -1.45+8.63256\,t, & t\le \dfrac{2\pi}{3},\\ 26.8687-4.43851\,t, & t>\dfrac{2\pi}{3}\end{cases}$$

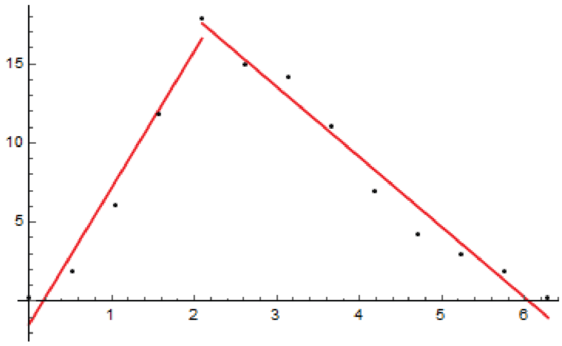

Figure 5. Piecewise linear least squares approximation

and then in a similar way like in 2.3 we get the third order trigonometric polynomial

$$\begin{aligned}\overline{T}_3(t)=\ & 8.04778-6.50084\cos t\\ &+3.31612\sin t-1.43032\cos 2t\\ &-1.04437\sin 2t+0.\ \cos 3t\\ &+0.05432\sin 3t.\end{aligned} \qquad (5)$$

CONCLUSION

Application of obtaining Fourier trigonometric polynomials after measurements are presented. They are based on the strategy of replacing a tabulated data with an approximating function which is easy to be integrated. Following the proposed approaches, harmonics of higher order could be calculated and spectrum of periodic function could be determined. The choice of approximating method and approximating function is not unique and it is not clear which one is the best possible. In the present work a variety of opportunities in this direction is proposed [2,5]. The ideas could be applied to other engineering studies.

Acknowledgement. This work is partially supported by the Technical University of Gabrovo under grant 2206C/2023.

REFERENCE

[1] M. Racheva, Applied Mathematics, Vasil Aprilov – Gabrovo, 2018.

[2] S. Chapra, R. Canale, Numerical Methods for Engineering, 6th edition, Mc.Graw Hill – High Education, 2010.

[3] D.W. Jordan, P. Smith, Mathematical techniques, 4th edition, Oxford, 2008.

[4] https://www.wolframcloud.com/

[5] https://www.wolfram.com/language/fast-introduction-for-programmers/en/