Annales UMCS Informatica AI 7 (2007) 27-36

Annales UMCS
Informatica
Lublin-Polonia
Sectio AI

http://www.annales.umcs.lublin.pl/

Algorithms for finding the maximal length quasiplateau interval of the experimental curve

Ewa Liśkiewicz*, Rafał Świetlicki, Jerzy Niewczas

Institute of Mathematics and Computer Science, The John Paul II Catholic University of Lublin, Konstantynów 1 H, 20-708 Lublin, Poland

Abstract

There exist some problems related to the determination of the quasiplateau region in experimental sciences. The paper describes two approaches to this matter which lead to two algorithms for finding the maximal length quasiplateau interval for discrete data. Numerical tests are presented.

1. Introduction

Problems involving finding some values based on analysis of the graphical representation of test results occur frequently in experimental sciences. Often is it important to find zero points, global and local extrema, saddle points and plateau regions of the curve. Determined values are utilized for specific purposes later on.

For example, mechanical stretching of individual polysaccharides with the use of atomic force microscopy (AFM) showed that glucose rings undergo forced conformation changes. That leads to the stepwise increase of polymer length, registered as a plateau of the curve showing the dependence between force and length [1].

Similarly, determination of potential oxygen flux in soil with the use of the voltammetric method is based on analysis of the current-voltage curve presenting the dependence between the current and the potential on the cathode, $I(U)$. When the oxygen flux magnitude is limited by the diffusivity of the medium, which manifests itself in the occurrence of a plateau in the $I(U)$ dependence, the calculated integer of the function $I(U)$ in the plateau area is converted into oxygen flux in the soil [2]. Because of the discrete character of

*Corresponding author: *e-mail address*: ewalisk@kul.lublin.pl

the experiment results, the plateau is called a quasiplateau. However, there is no precise definition of that term in literature.

Important problems, arising during such experimental data analysis, are to find out if the curve has a quasiplateau, and which interval of that curve can be recognized as the quasiplateau. Hence the necessity of developing effective algorithms in this area.

Figures 1 and 2 present the graphs of two voltammetric curves. In the opinion of an expert, the curve in Figure 1 does not have a quasiplateau, whereas that in Figure 2 has one.

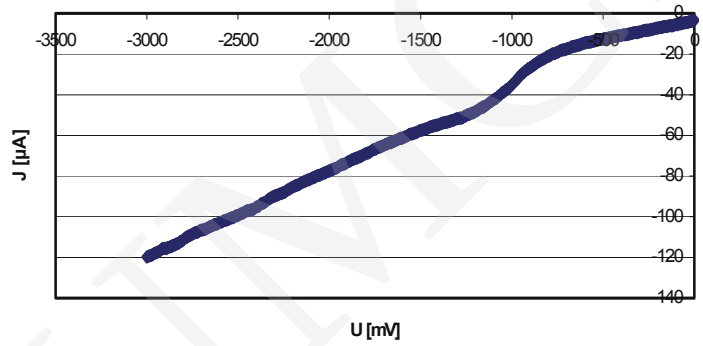

Fig. 1. Voltammetric curve without the quasiplateau interval

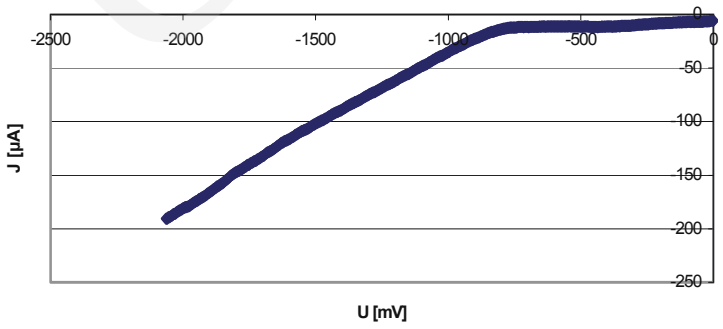

Fig. 1. Voltammetric curve with the quasiplateau interval

However, experts' opinions are subjective, based on their experiences. In many cases they are divergent, thus they cannot be considered to be fully adequate.

The goal of this article is to present two algorithms for finding the quasiplateau interval of the maximum length. The first algorithm (QP1) is based on searching for the longest sequence of arguments, for which the corresponding

values belong to a fixed interval. The second algorithm (QP2) consists in using the classical measure of variation, i.e. standard deviation.

2. Data and parameters

Let $K = \{0,1,\ldots,n-1\}$ be the set of indices. The input data for both algorithms are:

n – the number of points of the analyzed curve, $n > 2$,

(x_k, y_k), $x_k < x_{k+1}$, $k \in K - n$ pairs of real numbers which are the coordinates of measurement points

Let the parameters be:

ε_y – the non-negative, small enough real number, treated as tolerance of the function values dispersion on the quasiplateau interval,

δ_x – the positive integer, determining the minimum number of points belonging to the quasiplateau interval.

Taking as a parameter the minimum number of points δ_x, for which the selected interval can be designated as the quasiplateau interval, makes both algorithms more flexible. Whereas the parameter ε_y gives the possibility of arbitrary determination of the acceptable size of function values scatter on the quasiplateau interval, depending on the scattering level of the function values at all measurement points x_k, $k \in K$. Particularly, ε_y can take measurement errors into account.

For storing the vectors $X = [x_0,x_1,\ldots,x_{n-1}]$ and $Y = [y_0,y_1,\ldots,y_{n-1}]$ data structures are used that are well known from different programming languages, i.e. arrays indexed from zero. Additionally, there is introduced an array Z, for storing the values from an array Y in the nondecreasing order.

To assure the uniqueness of results, in the case of finding more than one quasiplateau interval, algorithms return the value of the longest one. Whereas, for many intervals of the maximum length, algorithms choose the first one located, i.e. the closest to the beginning of the coordinate system. In particular it means that:

- for two quasiplateau intervals $[x_i,x_j]$ and $[x_l ,x_p]$, for which $|j - i| < |p - l|$, $i < p$, the result is $[x_l,x_p]$,
- for two quasiplateau intervals $[x_i,x_j]$ and $[x_l,x_p]$, for which $|j - i| = |p - l|$, $i < p$, the result is $[x_i,x_j]$.

3. Algorithm (QP1) for finding the maximal length quasiplateau interval

The algorithm QP1 defines *the quasiplateau interval* as the interval $[x_{i(k)}\ x_{j(k)}]$ for which it is possible to find integers k, $i(k)$, $j(k) \in K$, $i(k) < j(k)$ fulfilling all of the following three conditions:

a) the interval $[y_k - \varepsilon_y, y_k + \varepsilon_y]$ contains the numbers $y_{i(k)}, y_{i(k)}+1, \ldots, y_{j(k)}$;

b) $j(k) - i(k) \geq \delta_x - 1$;

c) the difference $j(k) - i(k)$ is the biggest possible.

In any other case it is stated that the quasiplateau interval does not exist.

The search goes as follows:

for each integer $k \in K$, from all intervals in the form of $[x_{i(k)}, x_{j(k)}]$ (pair $(i(k),j(k))$ is any pair of integers that $i(k), j(k) \in K$, $i(k) < j(k)$) the interval is chosen which fulfills the conditions a) – c).

At the end, from all pairs $(i(k),j(k))$, $k \in K$, if at least one was found, that one pair is chosen for which the difference $j(k) - i(k)$ is the biggest, and the interval $[x_{i(k)}, x_{j(k)}]$ is taken as the *quasiplateau interval*. In any other case it is stated that the interval does not exist.

If for some indices p, q the equality $y_p = y_q$ occurs, then of course $i(p) = i(q)$ and $j(p) = j(q)$. Therefore, for the algorithm optimisation, while checking the condition (a) array **Z** is used, not array **Y,** and repetitive numbers are omitted.

Below is a fragment of program code written in C++ language which, based on the arrays **Y** and **Z**, determines the first, longest *quasiplateau interval* or discovers the lack of it. The arrays **Y** and **Z** are defined in Chapter 2. The variables **delta** and **eps** correspond to the parameters δ_x and ε_y. Symbol **P** denotes "the candidate" for the potential *quasiplateau interval* (named the **Plateau**), and the logical variable **exist** returns information about the *quasiplateau interval* existence.

```
for (int i=0;i<n;i++)
   if (Z[i]<Z[i+1]
   for (int j=0;j<n;j++)
      if (fabs(Y[j]-Z[i]<=eps)
         if (P.length==0 {
            P.first=P.last=j; P.length=1;
         }
         else {
            P.last++; P.length++;
         }
      else {
         if (P.length>Plateau.length)
            Plateau=P;
         P.length=0;
      }
if (Plateau.length>=delta)
   exist=true;
else
   exist=false;
```

4. Algorithm (QP2) for finding the maximal length quasiplateau interval

Algorithm QP2 defines *the quasiplateau interval* as the interval $[x_i, x_j]$, $\delta_x - 1 \le j - i \le n - 1$ for which the following two conditions are fulfilled simultaneously:

a) $St.dev.(y_i, y_{i+1}, \ldots, y_{j-1}, y_j) \le \varepsilon_y$, where *St.dev.* means standard deviation of the values in parentheses,
b) $[x_i, x_j]$ is the biggest interval for which the following implication is true: for fixed i, from $St.dev.(y_i, y_{i+1}, \ldots, y_{m-1}, y_m) \le \varepsilon_y$ it follows that $St.dev.(y_i, y_{i+1}, \ldots, y_{m-2}, y_{m-1}) \le \varepsilon_y$, for each m such that $i + 1 \le m \le j$.

In the first phase of its run, algorithm QP2 uses a method known from image processing – thresholding based on image histogram [3]. A histogram is a summary graph showing a count of data points falling in various ranges. The effect is a rough approximation of the frequency distribution of the data.

In the considered case factorisation will be presented in the frequency table **T** by the number of appearances of the value y_k, $k \in K$ which belongs to successive intervals with width d described by the following formula:

$$d = \begin{cases} 2\varepsilon_y, & \text{for } \varepsilon_y > 0 \\ (y_{\max} - y_{\min})/100, & \text{for } \varepsilon_y = 0 \end{cases}$$

Thus, the content of table **T** is as follows:

$$T[i] = n_i \text{, for } i=0,1,2,...,L$$

which means that in the i-th interval $[y_{min} + i \cdot d, y_{min} + (i+1) \cdot d]$ n_i of values y_k, $k \in K$ have been found;

L is described by the formula below:

$$L = Trunc\left((y_{\max} - y_{\min})/d\right) \text{ for } d \neq 0,$$

where *Trunc* means the operation of truncating the real number to an integer.
Width d assumes the value of zero only in the case when $y_{max} = y_{min}$, which means that the whole interval of experimental data is the searched *quasiplateau interval* (as a matter of fact it is the plateau interval).
In the next step of the algorithm, from table **T** only those intervals are chosen which contain the appropriate number of function values appearance, described by the parameter δ_x, and only for them the last stage of analysis is performed.
Below is a fragment of the program code written in C++ language, which finds, based on table **T**, the first, longest *quasiplateau interval*, or discovers the lack of it. The arrays **Y** and **Z** are defined in Chapter 2. Variables **delta** and **eps** correspond to parameters δ_x and ε_y, **Temp** denotes "the candidate" for the potential interval (named the **Plateau**), and the logical variable **exist** returns information about the *quasiplateau interval* existence.

```
M=0;
for (int i=0;i<L;i++)
   if (T[i]>=delta) {
      if (m==i-1|| i==0) {
         left=Z[0]+i*d; right=left+d;
      }
      else {
         left=Z[0]+(i-1)*d; right=left+2*d;
      }
      Temp.first=0; Temp.last=0; Temp.length=0; j=0;
      while (j<n)
         if (Y[j]>=left && Y[j]<right) {
            Temp.first=j; Temp.last=j; Temp.length=1;
            j++; k=j;
            while ((k<n) && (St_dev(Y,Temp.first,k)<=eps)) {
               Temp.last++; Temp.length++; k++;
            }
            if (Plateau.length<Temp.length)
               Plateau=Temp;
            else
               if (Plateau.length==Temp.length && Plateau.first>Temp.first)
                  Plateau=Temp;
            Temp.length=0;
         }
         else
            j++;
   }
if (Plateau.length>=delta)
   exist=true;
else
   exist=false;
```

5. Results

The results of effectiveness of both algorithms are presented using the example of the data set (VA) received from the experiments done in the Institute of Agrophysics, PAS, in Lublin, Poland. That data are coordinates of points of voltammetric curves (472 measurement samples) for which the density of oxygen flux in soil is determined [4].

Figure 3 shows the curve course and coordinates of both limit points – **Pmax** and **Pmin**. Those are also points with extremely different coordinates.

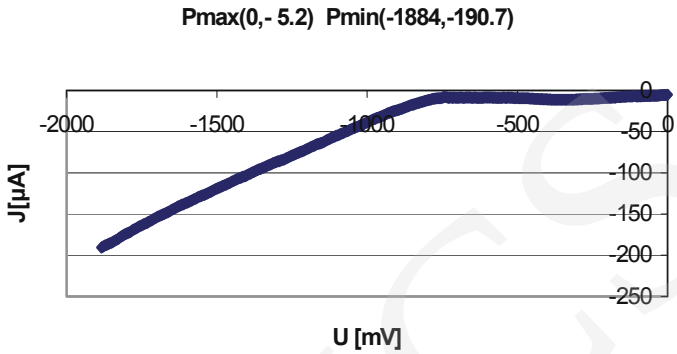

Fig. 2. Voltammetric curve (VA) of oxygen flux for data described with the symbol 45.4% (trielect) 568 (A1)

Tables 1 and 2 present the *quasiplateau intervals* and their lengths l, found with the help of both algorithms by arbitrarily chosen values of parameters δ_x and ε_y. The sign "–" denotes those cases when the algorithm does not find the quasiplateau interval.

Table 1. Algorithm QP1 results for VA data

ε \ δ	5	10	20	30	40
0	[-604;-588] $l = 16$	–	–	–	–
0.05	[-604;-588] $l = 16$	–	–	–	–
0.1	[-364;-308] $l = 56$	[-388;-308] $l = 80$	–	–	–
0.2	[-388;-308] $l = 80$	[-388;-308] $l = 80$	[-388;-308] $l = 80$	–	–
0.5	[-732;-464] $l = 268$	[-732;-464] $l = 268$	[-732;-464] $l = 268$	[-732;-464] $l = 268$	[-732;-464] $l = 268$
1.0	[-732;-400] $l = 322$	[-732;-400] $l = 322$	[-732;-400] $l = 322$	[-732;-400] $l = 322$	[-732;-400] $l = 322$

Tables 3 and 4 show, for the *quasiplateau intervals* described in Tables 1 and 2, the corresponding intervals of function values (in one case a single value).

Table 2. Algorithm QP2 results for VA data

ε \ δ	5	10	20	30	40
0	[-604;-588] $l = 16$	–	–	–	–
0.05	[-280;-252] $l = 28$	–	–	–	–
0.1	[-388;-308] $l = 80$	[-388;-308] $l = 80$	–	–	–
0.2	[-752;-512] $l = 240$	[-752;-512] $l = 240$	[-752;-512] $l = 240$	[-752;-512] $l = 240$	[-752;-512] $l = 240$
0.5	[-780;-436] $l = 344$	[-780;-436] $l = 344$	[-780;-436] $l = 344$	[-780;-436] $l = 344$	[-780;-436] $l = 344$
1.0	[-804;-160] $l = 644$	[-804;-160] $l = 644$	[-804;-160] $l = 644$	[-804;-160] $l = 644$	[-804;-160] $l = 644$

Tables 3 and 4 show, for the *quasiplateau intervals* described in Tables 1 and 2, the corresponding intervals of function values (in one case a single value).

Table 3. Function values for intervals from Table 1

QP1	Y(QP1)
[-604;-588]	-8,7
[-364;-308]	[-11.2;-11]
[-388;-308]	[-11.3;-11]
[-732;-464]	[-9.6 ;-8.6]
[-732;-400]	[-11 ;-8.6]

Table 4. Function values for intervals from Table 2

QP2	Y(QP2)
[-604;-588]	-8.7
[-280;-252]	[-10.4;-10.3]
[-388;-308]	[-11.3;-11]
[-752;-512]	[-9.1 ;-7.9]
[-780;-436]	[-10.3 ;-7.9]
[-804;-160]	[-12.4 ;-7.9]

6. Discussion

Let, for the given data set (X,Y), the intervals $[a_1, b_1]$ and $[a_2, b_2]$ ($a_1 \neq b_1$, $a_2 \neq b_2$) be *the quasiplateau intervals* found by the algorithms QP1 and QP2 respectively, with the fixed (equal for both algorithms) values of parameters δ_x

and ε_y. To compare those intervals the following similarity indicator w has been defined:

$$w = \left(1 - \frac{|a_2 - a_1| + |b_2 - b_1|}{2|x_{n-1} - x_0|}\right) * 100\% .$$

It is obvious that $x_{n-1} \neq x_0$ (data set (X,Y) cannot consist of one point). The indicator $w > 0$, because it follows from the definition that both intervals include more than one point. The indicator w is the bigger, the closer both intervals are to mutual cover; the biggest value (100%) is reached only in the case when both intervals are identical ($a_1 = a_2$ and $b_1 \neq b_2$).

Table 5. Comparison of found quasiplateau intervals using the similarity indicator w

QP1	QP2	*w*, %
[-604;-588]	[-604;-588]	100
[-604;-588]	[-280;-252]	82.48
[-364;-308]	[-388;-308]	99.36
[-388;-308]	[-388;-308]	100
[-388;-308]	[-752;-512]	84.93
[-732;-464]	[-780;-436]	97.98
[-732;-400]	[-804;-160]	91.72

It is obvious that if $\varepsilon_y = 0$, then both algorithms will determine the same interval (if it exists). It is easy to predict that the bigger tolerance of ε_y will be accompanied by the higher effectiveness of quasiplateau interval found for all accepted values of δ_x (Tables 1,2). As a result, the intervals found are getting longer and longer. It is symptomatic that the algorithm QP2 in all cases finds *quasiplateau intervals* which are longer or equal to those found by the algorithm QP1. It is worth mentioning that for $\varepsilon_y = 0.05$ and $\varepsilon_y = 0.2$ the intervals found by both algorithms are separate. For the biggest values of ε_y (equal to 0.5 and 1) the intervals found by QP1 become subintervals of those found by QP2.

Conclusions

Both proposed algorithms give the possibility of finding *the quasiplateau interval* for a data set of points (X,Y), if the interval exists, with the given tolerance ε_y and for a given minimum number of points δ_x in the interval. Tests show a high convergence of the results given by both algorithms, although the QP2 algorithm is more liberal than QP1 (finds broader intervals). This is an effect of using more liberal measurement of data scattering in the definition of *quasiplateau interval*.

The two algorithms do not exhaust diverse methodological approaches to the issue of finding *the quasiplateau interval*. The authors will continue their research in that direction, taking into consideration also subjective opinions of experts, as the one of the possible methods of comparison of the obtained results.

Acknowledgements

The authors would like to thank doc. dr hab. Andrzej Bieganowski from the Institute of Agrophysics, PAS, in Lublin for releasing the experimental data (and for all valuable remarks concerning the results).

References

[1] Marszałek P., *Chemical identification of individual macromolecules by means of atomic force microscopy*, Proceedings of XXXVI Congress of Polish Physicists, Toruń, (2001), in Polish.
[2] Bieganowski A., *Algorithm for the determination of quasiplateau of voltammetric curve of oxygen reduction*, Acta Agrophysica, (2002) 72, in Polish.
[3] Malina W., Smiatacz M., *Methods of digital image processing*, Akademicka Oficyna Wydawnicza EXIT, Warszawa, (2005), in Polish.
[4] Bieganowski A., *Metrological aspects of determination of potential oxygen flux density in soil*, Acta Agrophysica, (2005) 118, in Polish.