MNBIC Reference Guide

I. Introductory notes

The minimum perimeter polygon (MPP) has been studied mostly within the image processing applications and computer processing of digitized contours. Among the pioneering articles certainly belongs one published by J. Sklanski and col. [1] and since then dozens of interesting approaches and algorithms were designed. It is worth of mentioning the works done by Slovak researchers Sloboda and Zaťko, too - see e.g. [2] . I had taken a privilege to work with them for some time period almost three decades ago and they aroused the interest of the topic to me.
However, the approach here is authentic and is based largely on the elastic fiber stretching simulation. Minimal Normed Built-In Cirquits on the rectangular grid (further MNBIC) take over the concept of MPP (though not restricted to the closed curves and the clockwise rotation).

II. Motivation

Here are four fields of interest where the solution could be exploited.

1. Digital image processing and pattern recognition

   MPP and consequently MNBIC belong to the means where the digitized figures are represented by the external characteristics – its border. This can be useful for computer processing and image recognition. The advantage consists in the fixed length of the MNBIC regardless of the starting point and rotation for the fixed inner and outer contour border of the closed input curves. Besides the unique value of the length as the shortest length inscribed into the cellular contours I suppose that the number of vertices linked to this length for MPP or MNBIC having the origin in the inner and outer countours is also minimal.

   However, it is worth of mentioning, that there can be more than one representation of the MPP or MNBIC for a given figure but each of them must have the same length and number of vertices . Otherwise some of the variants should not be entitled for MPP or MNBIC qualification when we estimate the final result of the calculation. Further we will operate with the MNBIC term and everything for the closed input curves valid to MNBIC will be valid equally to MPP.

2. Approximation of the digitised curves on the plane

   Finding the MNBIC on rectangular grid can be useful as the alternative method for an approximation of explicit and implicit functions where their discrete values are at disposal. In such cases the rectangular grid basis at first should be taken into the consideration. Where the discrete values of functions are at disposal, the horisontal (Dx) grid side is recommended to be equal to equidistant step for arguments.

   In fact, the MNBIC technique can be applied to wide spectrum of figures regardless of they are or not are expressed in an analytical form.

3. Data compression techniques

MNBIC technique can be strightforwardly used for data compression and storing the big volumes of data for digitised figures , particularly their countours. The advantage is the inclusion of fault tollerance norm (see below Dx and Dy role).

4. Relation to Linear optimization calculation

Without going to the theory or details of the linear optimization methods let me consider this example (not necessarily simple) :

The task is to design the MNBIC inside the rectangular cellular contours of the digitized figure (contours represented by the inner and outer border). Besides the minimal length for the output polygonal line we require that MNBIC passes through any contour cell or at least one point of contour cell.

Furthermore we have two rolls of tapes in stock , in blue and red color . Blue tape will be used for the MNBIC edges where at least one vertex belongs to the inner grid border , red tape will be used for the MNBIC edges where both vertices belong to the outer grid border and only to outer grid border. The sequence of the adjacent edges in the MNBIC will not be cut but unrolled en block if the same color is assigned to the edges.

Some edges in MNBIC can be shared (not to confuse with the input inner edges where it is not possible). In such cases no double tapes of the overlapped part are needed.

Some questions like below can arise :

a: What is the shortest length of the sum of tapes inscribed into the cellular countour of the figure ?

b : What are the longest tapes for each color ?

c: If overlapped edges in the MNBIC exist - how much length we can spare by excluding double counting for overlapped parts ?

III.Characteristics of algorithmic performance

1. Low topological restrictions on the curves (no splitting to the segments due to the topological constraints needed)

Some approaches for MPP calculation use to split the continuous figure into several segments , process them and after that merge them . This is not the case unless we do not take into account the dynamic pipeline processing . A determining factors consist in four condititions thus allowing us not to take care of additional topological attributes of the input figure.

- no two edges are intersected
- no two edges are partly or entirely shared
- no edge can touch the other edge by the vertex
- no two different vertices have the same coordinates

-

2. Minimal length and minimal number of vertices (these are subordinated to the minimal length) for output polygonal lines. This is a lucid demand and the flagship characteristic of the MNBIC calculation.The integral part for the length criteria is a norm as follows :

   for any point (x,y) of the input contours of the figure (IF) there exists at least one point $(x^*,y^{*)}$ of the MNBIC (and vice versa), where :

   abs $(x - x^*)$ <= Dx and abs $(y - y^*)$ <= Dy whereby

   Dx resp. Dy is horizontal resp. vertical side of the grid (raster) and

   x, y € IF, $x^{*,}$ y^* € MNBIC

3. Low complexity of calculation

   The average calculation complexity is linear O(n) operations where n is the number of vertices for the input polygon. In a worst case a crude estimate is the $O(n^2)$ operations but this seems to be very pesimistic and the question arises to replace the complexity in worst case by O(nlogn) operations. This is an open question till now.

4. Dynamic processing

   This feature enables to process the infinite chain of vertices in the pipeline regime. This can be suitable as needed to pick up data for image recognition and their comparision using the mobile scanning device.

   In such a case the coming stream to device is processed in continuos regime and simultaneously trims the fragment. This fragment is postprocessed simultaneously but independently to the main processing of the coming input vertices. The postprocessing affirms the final form of fragment to be ready to be merged again to the main stream.

   It is worth of mentioning that there can occure simultaneously several fragments in a postprocessing phase along with the main live stream and / or some are ready to be merged. Moreover, the cutting for fragments is done only by the vertices they are entitled to be " connectors " to adjacent fragments.

IV. Pocket reference guide - Handling your own drawings :

To perform your own drawing is possible by the executive file DrawYourMNBIC. Start to draw by positioning a cursor on the screen . You reach any place by the sequence of L/l (left), R/r (right), U/u (up) and D/d (down) buttons and fix the place by the * (asterisk) button. It is upon you if you will use sketch grid for a better navigation or not. Then you draw your own figure by pressing bottons mentioned above leaving one after another behind footmarks. When you change a direction in such a way that both coordinates x,y of the present footmark will differ from the last recorded vertex a new vertex will be recorded. If you press any button except mentioned above and '\r' (enter), no action will be done. In case of '\r', you finish the program and the MNBIC will not be evaluated. If you commit one of four drawing violations outlined in III-1, the last edge will be deleted and you can attempt the another but correct cursor action. You can also delete your last edge by your own choice (and in the reverse run next to last repeatedly) if you will share the edge in opposite direction. You can leave also the curve open if you push subsequently twice X/x. Here the system (toy version) allows you to collect maximally 1000 vertices, then the curve will be automatically frozen (not closed) and MNBIC evaluated. Prior to this action a notice linked to the vertex in a sequence number 994 will appear.

The result will be affected by the outer border in the lightcyan colour. Consequently, each edge of the MNBIC is determined by the vertices from the inner (basic contour) and/or the outer border.

The final MNBIC is displayed by the red and blue colours in the alternate manner for the adjacent edges to better distinguish lines. In case where the edges (or part of them) are shared in both directions a yellow colour is used for the illustration. To make the acces to any of your previous MNBIC later you automatically put the result data from the file DrawYourMNBIC into the file DataMyDrawing and to invoke it by DisplayMyDrawing executive file. In this way you can create an archive of your own drawings and return back to them later. Do not forget that your last output data file will have the name DataMyDrawing too, so you should rename it before using the same name to other data file. You can also switch among modes for your drawing by B/b, P/p, M/m and C/c analogically to Demo illustration. There are vertices of MNBIC prompted by crosslets in two colors in PIN mode. The crosslets in lightmagenta color belong to inner border vertices whereas the crosslets in lightcyan color belong to outer and only outer border vertices. Bear in mind that some inner and outer vertices can be overlapped. In such case lightmagenta color for crosslet will be assigned.

Some useful notes :

You do not need to care about the direction you start to draw. But keep in mind that implictly your drawing will be evaluated in a clockwise manner with the respect to “ any vertex of fictitious convex hull of polygon “. In practise such vertex will be “ most right-down“ or “ most left-up“ vertex of your closed polygon. Then, If you take into account a predecessor of this “convex hull vertex“ and consequently the ancestor of this vertex, the outer border (and the related vertices for outer border) will be created on the left side of these two lines in clockwise manner.

For the open curve the position of the outer border will be derived from the most right - down position of the contour vertex and the clockwise rotation related to the predecessor and the ancestor of this vertex. The border will be situated on the left side . One could reclaim that the terminal vertex is the most right – down alone. In such a case move the vertex index forward resp. backward and apply the same reason .

The user can change a clockwise rotation of MNBIC to anticlockwise rotation (and vice-versa) by pressing the '@' button within the both files DrawYourMNBIC and DisplayMyDrawing. It is obvious that you obtain in general two diffent results. Left side in this case is defined with the respect of anticlockwise rotation. To use the DisplayMyDrawing file properly , you are obliged to switch the rotation earlier in DrawYourMNBIC file to download data.

In case of open curves the starting and ending vertices will always belong to the inner border (rather to say to your white colored drawing curve) for both rotations . To avoid the ambiguity (mainly in communication) let me recommend to use postfix to MNBIC related to the rotation. MNBIC-CR resp. MNBIC-AC will represent MNBIC in clockwise rotation resp. anticlockwise rotation.

Finally, I believe that all exe.files should properly act under Windows 7 and higher version.

Acknowledgements :

Couple of personalities interested in the topic (including the relax and/or intensively using Pythagorean theorem to check the lengths of the alternative fragments of MNBIC) provided hundreds of tests to verify the results and/or helped me with a web presentation to be prepared. My gratitude belongs to them.

Imrich Móri Kalinkovo, May 2022

References

[1] J. Sklanski, R. L. Chazin, B. J. Hansen
Minimum perimeter polygons of digitized silhouettes
IEEE Transactions on Computers, 21 (3) (1972), pp.260 – 268

[2] F. Sloboda, B. Zaťko, J. Stoer
On approximation of planar one-dimensional continua
R. Klette, A. Rosenfeld, F. Sloboda (Eds.), Advances in Digital and Computational Geometry (1998), pp. 113 – 160