New method of creation data for natural objects in MRDB based on new simplification algorithm

Krystian Kozioł, Stanisław Szombara

AGH University of Science and Technology, Faculty of Mining Surveying and Environmental Engineering, Department of Geomatics

Abstract: In the paper, authors present a prototype of new simplification algorithm which first divide the original polyline into surjection parts and then converting into this parts in to curve H(x) and then located points depending on the scale of the generalised map. This method is useful for simplification of the natural objects. Additionally, the extreme points of the curve, which are in accordance with the standard of the drawing recognition (recognisability)[1] (Chrobak 2010) are taken into consideration while locating. Presented algorithm of simplification use as threshold parameter the standard of the drawing recognition, which is independent of the user. Moreover the new algorithm increases the automation of the polylines simplification process and increases the scale range, which is particularly important in the Multiresolution/ multiRepresentation Data Base (MRDB) applications.

Introduction

One of the main tasks of cartography in the 21st century is visualisation of topographical objects in Multiresolution/multiRepresentation Data Bases in various scales. This task is closely related to digital cartographic generalisation. Mackaness and Ruas (2007) point out four reasons for the generalisation process to be so difficult to automate. They state, that generalisation is more than just a geometric process based on the analysis of main geometric types. According to Sarjakoski (2007), creating MRDB means using generalisation model into account the following criteria:

- most frequently used representation of objects
- requirements considering actualisation of objects
- level of automation, that can be achieved in the process of creating the representation of objects, while obtaining the representation of objects at lower level of detail.

In the considerations raised by the above mentioned authors, problems of generalisation model and automation in terms of MRDB functioning are continuously discussed. Sarjakoski (2007) suggests the range of necessary research including: modelling of data, automatic instruments for creating MRDB, as well as detecting discontinuities and conflicts in MRDB.

[1] The standard of the drawing recognition (recognisability) was defined by (Chrobak 2000) on the basis of the least drawing dimensions provided by (Saliszczew 1998). The length of shorter side of the recognisability triangle is in agreement with the measure of the cartographical norm set by National Map Accuracy Standards – NMAS(Longley *et al.* 2006). The standard of the drawing recognition is determined by a black colored elementary triangle with the following dimensions: line width 0.1mm, shorter side a = 0.5 M (mm) and base $b \in [(0{,}5 - 0{,}7)M]$ [mm].

According to the authors of this article, the degree of process automation depends on: data orderliness, operators interoperability, unambiguity of results and also possibility of results verification. For the simplification operator the correct order of data is a superior element, significant while treating the arising conflicts. Unambiguity of simplification results is achieved by determining the process parameters basing on the standard of the drawing recognition and not knowledge of the user. Verification of the results is possible only if standard which were used retain the role of comparing a pattern to a case. Number of conflicts occurring in the simplification process is an additional element of verification.

Simplification operator

The spatial data generalisation models under elaboration show how complicated the process is. For instance, in the Brassel-Weibel model (Brassel and Weibel 1988), there is a stage of modelling called process and parameter recognition, in which the proper generalisation operators are created. The same approach is used in the McMaster and Shea model (Shea and McMaster 1989), whose authors defined the generalisation operators, and so did McMaster and Monmonier in their model (McMaster and Monmonier 1989). The authors underlined the necessity to use different operators for generalisation of different objects. Claimed to be one of the basic operators, the simplification operator is used at the very beginning of the modelling process, dealing with the linear and surface objects. After long research on simplification operator functions, it is still impossible to point out the algorithm which would be optimal in this application. It is claimed that the first simplification algorithm is the one created by Perkal (1966) and Lang (1969).

However most of the algorithms were created in the second half of the 20th century, including one of the most popular, namely the Douglas-Peücker algorithm (Douglas and Peucker 1973). In general, the basic parameter of the Douglas-Peücker algorithm is the length determined by the user. A different approach was proposed by Visvalingam and Whyatt (1993). Their algorithm bases on the determined surface of a triangle. As far as the analysis of the original polyline is concerned, one of the most advanced algorithms seems to be the one created by Wang (1996). He has proposed examining the original polyline and adjusting the algorithm to its shape.Li (2007), in his research on generalisation process, has divided the simplification algorithms into the point-reduction and scale driven generalisation algorithms. The first group consists in selection of critical points of the polyline and subsequent elimination resulting from the set target scale. This group is further divided into: independent point-based procedures, local processing procedures, extended unconditional local processing procedures, extended conditional local processing procedures and global procedures (McMaster 1991). The global algorithms are represented by: Douglas-Peücker algorithm, Visvalingam-Whyatt algorithm, Wang and Chrobak algorithm Chrobak (2000). Dyken, Dæhlen, and Sevaldrud (2009) presented a method for simultaneous simplification of a collection of piecewise linear curves. This method based on triangulations removing line segments from the piecewise linear curves without changing the topological relations between the curves.

Described algorithms of simplifying have one shared feature they are eliminating points from the original polyline according to the procedures determined by the algorithms. Whereas, in

fact, the process of simplification is such a choice of vertices from the original polyline, that the similarity of shape, location and measure of the generalised polyline are retained (Fig. 1). The value of parameter in the algorithms depends on the knowledge and experience of the user.

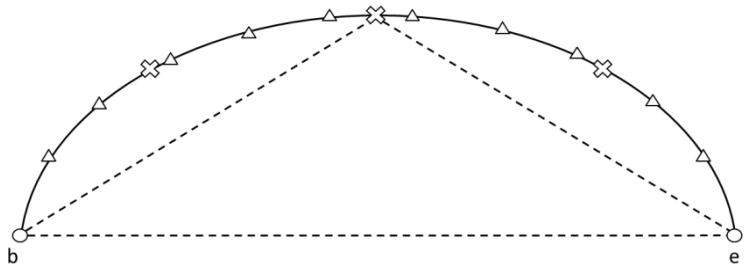

Figure 1. Different location of the object vertices in various scales.

The authors present a new prototype of simplification algorithm for polylines in the paper. The projected simplifying method is suitable for the objects of natural (smooth) character, such as rivers, lakes and coastline.

The stages of the algorithm are:

- locating a characteristic vertex on to polyline which divide this polyline in to parts which are surjections,
- transformation of the polyline into a curve with an infinite number of points, using the Hermite polynomial interpolation,
- determination of the extreme points of the original polyline and their hierarchy, satisfying the standard of the drawing recognition,
- selection of new intermediate points of the curve between neighbouring extreme points and at distances satisfying the standard of the drawing recognition.

In result of the simplification of a polyline, the shape and location of both the original open polyline and the simplified one are preserved within the accuracy limits set by the recognisability norm. Furthermore, due to the recognisability of the new algorithm, the value of the simplification parameter is not determined by the user and thus it is possible to obtain an unambiguous result. Both, the unambiguity of result and the fact that the user does not have to determine the parameter, increase the process automation. Thus, the new solution is more a method than just an algorithm.

The new simplification algorithm

The flowchart presented below (Fig. 2~~5~~) was implemented in MATLAB using its inner programming language. MATLAB was chosen due to the possibility to use its existing libraries, which include the Hermite polynomial interpolation. The algorithm is based on:

- a loop search for global extreme of a function satisfying the standard drawing recognition,
- a loop in which the intermediate points are inserted in agreement with the standard drawing recognition.

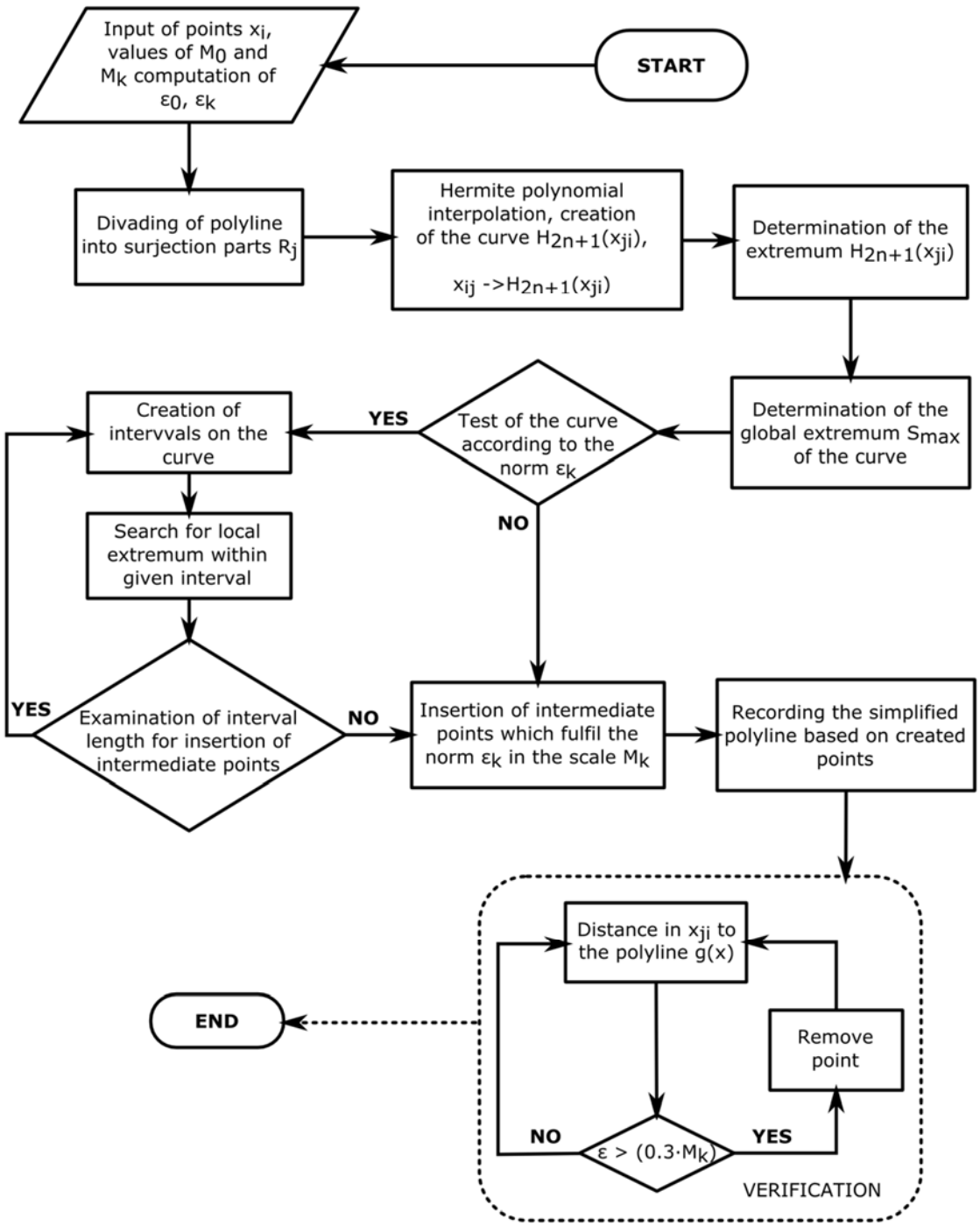

Figure ~~5~~2. The flowchart of the simplification algorithm.

At the beginning the user loads the file (ESRI Shapefile format) which contains source data. He also enters the scale denominator of the source data and the destination scale denominator M_0 and M_k, which determine the values ε_{01} i ε_{02}. Next the polyline is divided into the surjection parts. Only then the discrete data can be (with use of interpolation) expressed as a spline function which satisfies the condition 18. The process of dividing the polyline into

surjection parts is described in the next chapter. The transformation of a polyline into a curve takes place for the surjection parts and allows us to determine the extremum of this function. This enables us to check the recognizability norm (Equation 23). If the triangle beginning point-extremum-end point satisfies the condition of the norm, then the other vertices of the curve in the given interval may be recognizable. These vertices are searched for as the further determination of extremes is being carried out and the standard of the drawing recognition is checked. As a result a new polyline is created. This polyline undergoes verification which is described in details in the further chapters of this paper.

Locating the characteristic vertex on to polyline (creation of surjection parts)

Characteristic point is an important concept in many disciplines such as computer vision, image processing, pattern recognition, computer graphics, and geospatial science. For example, in computer vision and pattern recognition, the concept of critical points is used in algorithms for feature extraction, shape recognition, point- based motion estimation, and coding. In geospatial science it is used for data compression, for line caricature, and misleadingly for multi-scale representation of lines (Li 1993).

Unambiguous interpolation performed with use of the Hermiet method requires vertices and edges to satisfy the condition of surjection and therefore any polyline has to be divided into parts. Those parts have to satisfy the following condition: in local rectangular Cartesian coordinates system with origin in the first node of a R_j part and x-axis directed towards the last node, all the vertices and edges of a part must be a surjection and satisfy the condition:

$$\forall y \in [y_{min}[y_i, R_j], y_{max}[y_i, R_j]] \exists x \in [0, x_{max}[x_i, R_j]] : f(x) = y, \quad (1)$$

where:

- $x_{max}\lfloor x_i, R_j \rfloor$- maximum x-coordinate (abscissa) of the vertices of R_j,
- $y_{min}\lfloor y_i, R_j \rfloor$- minimum y-coordinate (ordinate) of the vertices of R_j,
- $y_{max}\lfloor y_i, R_j \rfloor$- maximum y-coordinate (ordinate) of the vertices of R_j,

and the number of such parts should tend to minimum.

Such parts will be called unambiguous and the vertices which determinate them as characteristic points (Li 2007). In the Figure 3~~2~~ a polyline with its unambiguous parts is presented.

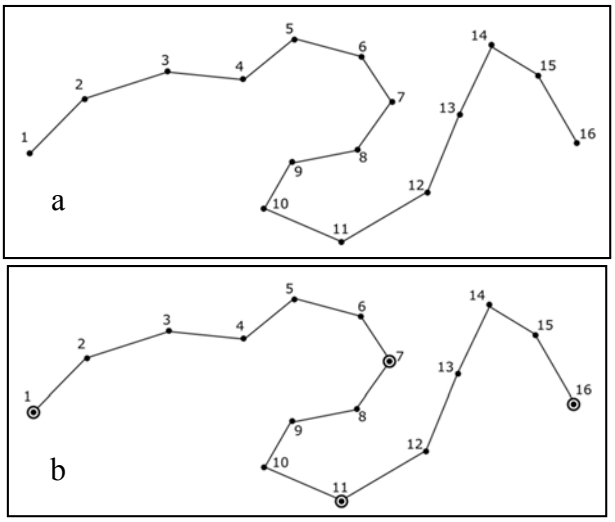

Figure 3~~2~~. Original polyline a) and polyline divided into unambiguous parts b).

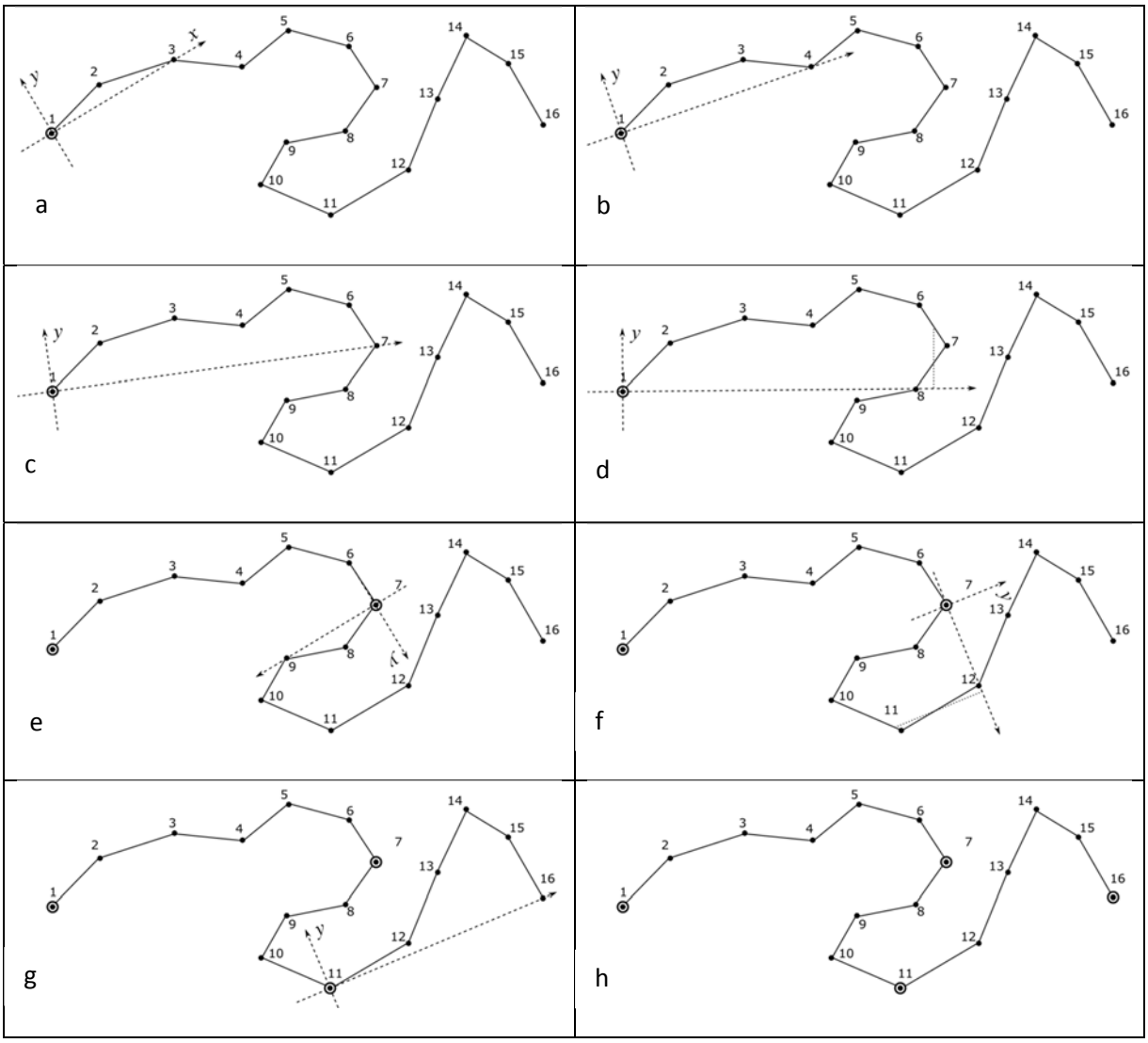

Figure ~~3~~4. Determination of unambiguous parts (described in text).

The algorithm of unambiguous parts determination was described with use of notation based on the (Molenaar 1998) notation. For the polyline P_i, the set of its vertices is given as:

$$V_{P_i} = \{v_1,\ldots,v_j,\ldots,v_n\}, \tag{2}$$

and the set of its edges as:

$$E_{P_i} = \{e_1,\ldots,e_j,\ldots,e_n\} = \{\{v_1,v_2\},\ldots,\{v_k,v_{k+1}\},\ldots,\{v_{n-1},v_n\}\}. \tag{3}$$

The Direction of $P_{\cancel{1}i}$ ~~(Figure 2a)~~ is specified by:

$$Dir[P_i] = \{v_1,v_n\}. \tag{4}$$

Let $R_i(P_j)$ be an unambiguous part and V_{Ri} its set of vertices:

$$V_{R_i} = \{v_1,v_2,\ldots,v_k,\ldots,v_{n-1},v_n\}, \tag{5}$$

whose coordinates can be treated as attributes:

$$x_i = X[v_i] \text{ oraz } y_i = Y[v_i], \tag{6}$$

The figure 4~~3~~ depicts the algorithm of unambiguous parts determination. This algorithm begins with creating R_1 and assigning of vertices v_1, v_2 to it. Next, the following vertices are examined one after another and added to R_1 as long as the condition (1) is satisfied. If the condition (1) is not satisfied a new part is created. The algorithm examines all the vertices of the polyline. Figure 4~~3~~a shows adding of v_3 to R_1. The x-axis of local cartesian coordinate system (originating in v_1) is directed towards this vertex.

For $R_1(P_j)$ (at Figure ~~3~~4a) containing v_3 the following can be stated:

- the set of its vertices according to (2)

$$V_{R_1} = \{v_1,v_2,v_3\}, \tag{7}$$

- the set of its edges according to (3)

$$E_{R_1} = \{e_1,e_2\} = \{\{v_1,v_2\},\{v_2,v_3\}\}, \tag{8}$$

- its direction according to ~~(4)~~

$$Dir[R_i] = \{v_1,v_3\}, \tag{9}$$

- the range of its coordinates (in local system):
  - $x_{\min}[x_i,R_1]$- minimum x-coordinate (abscissa) of the vertices of R_1,
  - $x_{\max}[x_i,R_1]$- maximum x-coordinate (abscissa) of the vertices of R_1,
  - $y_{\min}[y_i,R_1]$- minimum y-coordinate (ordinate) of the vertices of R_1,
  - $y_{\max}[y_i,R_1]$- maximum y-coordinate (ordinate) of the vertices of R_1,
  - relationships of the beginning and end of the part:

$$BeginR[R_1] = \{v_1\}, \tag{10}$$

$$EndR[R_1] = \{v_3\}. \tag{11}$$

Upon adding v_3 and change of the orientation of the local coordination system the condition (1) is examined, which can be noted as:

$$\forall y \in [y_{min}[y_i, R_1], y_{max}[y_i, R_1]] \exists x \in [0, x_{max}[x_i, R_1]] : f(x) = y \ . \tag{12}$$

If the condition (12) is satisfied, R_1 will be enlarged by v_3 (as depicted in fig. 43a) and algorithm will move on to the next vertex. Figures 43b, 34c, 3e show the situation in which the condition (1) is satisfied and the algorithm moves on to the next vertex, whereas the figure 43g presents the case where the condition (1) is satisfied but algorithm finishes its work. If the condition (1) is not satisfied (fig. 43d, 43f) then:

- R_1 is saved without the examined vertex,
- a new origin of a local coordinate system id determined in $EndR[R_i]$,
- a new part R_{i+1} is created with beginning in $EndR[R_i] = BeginR[R_{i+1}]$.

After the algorithm has been performed (fig 42h) we obtain:

- the final set of unambiguous parts of the polyline:

$$R_{P_i} = \{R_1, R_2, R_3\}, \tag{13}$$

- the set of vertices which determine parts (characteristic points (CP)) is described as:

$$CP_{P_i} = \{v_1, v_7, v_{11}, v_{16}\} = \{BeginR[R_1], BeginR[R_2], BeginR[R_3], EndR[R_3]\}. \tag{14}$$

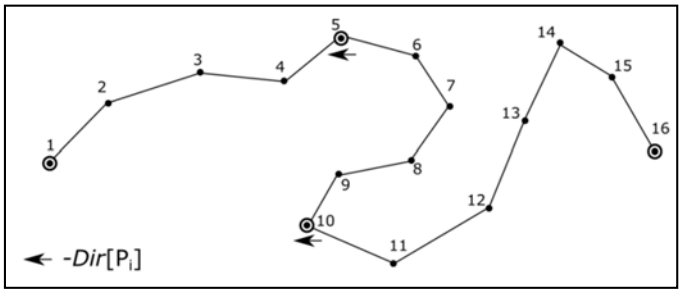

Figure 54. Characteristic points of the polyline determined with use of algorithm performed reverse to its direction.

It can be easily observed that, according to the direction in which we perform the algorithm, whether from the beginning to the end of a polyline (Fig. 43h) or reverse (Fig. 54), the number of the obtained parts (as well as the number of characteristic points) may differ. For the polyline we choose this direction which has the smaller number of R parts $R_i(P_j)$. If both directions give the same number of parts we take the parts (and of course the direction) on which the characteristic points are placed more regularly. This regularity is measured by coefficient of variation of parts' lengths.

Creating curve using 3rd degree Hermite Interpolation

The first step of the algorithm is converting polyline into a curve, continuous within the given interval, with use of a chosen interpolation method (Fig. ~~5~~2). The chosen interpolation method should satisfy the following conditions for a curve:

- the curve should pass through all the points of a polyline – f(x) = H(x),
- the local extreme of a polyline should be preserved – f'(x) = H'(x).

These conditions are satisfied by the Hermite method. We know the number of points and the values of their coordinates, for which exists the function of class $C^1 \in$ [a, b]:

$$f(x) \in C^1[a,b] \wedge [x_0, x_1, x_2, \ldots, x_i, \ldots, x_n] \in [a,b], \tag{15}$$

where: $x_i \neq x_{i+1}$ for i = 1,2,3,…,n-1.

We use the Hermite polynomial, compatible with $f(x_i)$ and with $f'(x_i)$ at points x_i for i = 0,1,2…,n, furthermore the polynomial is of no greater degree than 2n+1.

The form of the polynomial (Boor 1978) is as follows:

$$H_{2n+1}(x) = \sum_{j=0}^{n} f(x_j) H_{n,j}(x) + \sum_{j=0}^{n} f'(x_j) \hat{H}_{n,j}(x), \tag{16}$$

where:

$$H_{n,j}(x) = \left[1 - 2(x - x_j) L'_{n,j}(x_j) L^2_{n,j}(x)\right] \wedge \hat{H}_{n,j} = (x - x_j) L^2_{n,j}(x),$$

$L_{n,j}$ is a jth coefficient of the Lagrange polynomial of nth degree.

If f(x) $\in$ C 2n+2 [a, b], then there is the following relation between the function f(x) and the Hermite polynomial:

$$f(x) - H_{2n+1}(x) = \frac{(x - x_0)^2 \ldots (x - x_n)^2}{(2n+2)!} f^{2n+2}(\xi), \tag{17}$$

where: $a < \xi < b$

In order to be able to claim that Hermite interpolation satisfies the conditions of an unknown interpolation function (preserves its shape), it is enough to prove that:

$$\forall_i H_{2n+1}(x_i) = f(x_i) \wedge \forall_i H'_{2n+1}(x_i) = f'(x_i), \tag{18}$$

where: i = 0, 1, 2,…, n.

Thus, we should prove that expressions $H_{n,j}$ and $\hat{H}_{n,j}$ determined by equations (16) satisfy simultaneously the following four conditions:

$$H_{n,j}(x_i) = \begin{cases} 1 & for \ j=i \\ 0 & for \ j \neq i \end{cases} \tag{19}$$

$$\forall_i \frac{d}{dx} H_{n,j}(x_i) = 0, \tag{20}$$

$$\forall_i \hat{H}_{n,j}(x_i) = 0, \tag{21}$$

$$\frac{d}{dx} \hat{H}_{n,j}(x_i) = \begin{cases} 1, & for \ j=i \\ 0, & for \ j \neq i \end{cases}, \tag{22}$$

what has been proven by de Boor in his book (De Boor 1978).

After converting the polyline into a curve (with use of an interpolation method) we obtain the most probable outline of shape of the original object. The next stage of the algorithm is to arrange the intermediate points on the original curve, depending on the scale.

Arrangement of points on the original curve according to the standard drawing recognition

All the simplification algorithms are based on the geometrical condition, which should be satisfied for the vertices remaining after the simplification process. By use of the proposed algorithm, after the interpolation process, we obtain a curve with unlimited number of vertices defined by function H(x). It is therefore necessary to choose only the vertices, which will be characteristic of the outline and location of a shape in the target scale.

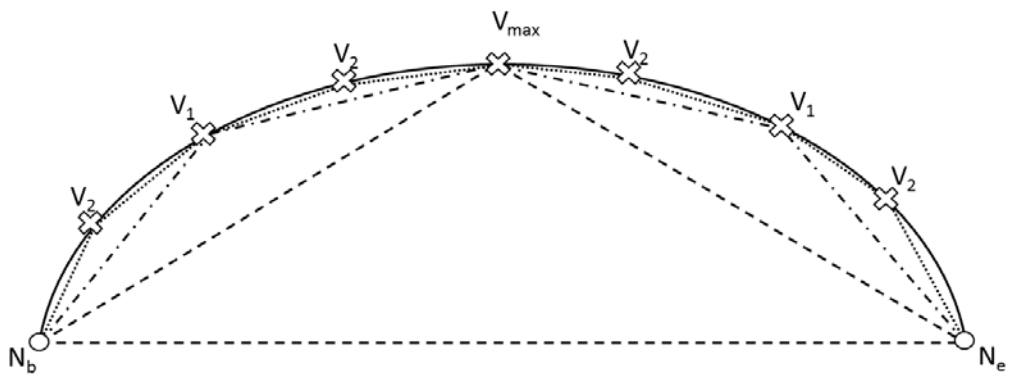

Figure 6. Choosing the vertices of a curve.

Checking of the elementary triangle condition for the triangle (N_b, V_{max}, N_e) enables deciding on the recognisability (the standard of the drawing recognition) of the vertex, which is the global extreme V_{max} (Figure 6). In such a way it is determined if the triangle will be represented in the analysed scale. Then, in the same way, the algorithm checks all the local extremes on the basis of the adequate triangles (Figure 6). After determining the recognisability of the local extremes, the algorithm arranges the remaining intermediate vertices. In order to distinguish points of the original curve (16) for preparation of a map in the scale 1:M_k we have used:

- the recognisability based on the elementary triangle (Chrobak 2010):

$\varepsilon_{01} = 0{,}5[mm]*M_k \wedge b \in [0{,}5mm - 0{,}7mm)*M_k]$,

$\varepsilon_{02} > 0{,}5[mm]*M_k \wedge b \in [0{,}4mm - 0{,}5mm)*M_k]$ (23)

where:

ε_{01} i ε_{02} - lengths of the shorter sides of the triangle (Figure 6~~3~~),

b – lengths of the base of the triangle,

M – map scale denominator,

- *f* the extreme s_m *(m=1,2,...,l)* of the function (15) and the polynomial (16), in agreement with recognisability,

The presented conditions for arrangement of the new intermediate points on the curve (16), enable us to:

- verify the interpolation, by comparison of the extreme points of the polyline (15) with the extreme of the original curve (16). The result of the comparison is identity, which proves that the Hermite polynomial interpolation is correct,
- determine unambiguously – with no operator intervention needed – extreme points of the polyline under simplification into the map scale 1:M_k (k = 0,1,2,…,n), provided both standard of the drawing recognising and the original hierarchy of critical points are preserved,
- create, in the map scale 1:M_k (k = 0,1,2,…,n), intervals based on the determined extreme points of the original curve, within which new intermediate points will be chosen at distances compatible with the standard of the drawing recognition suitable for the scale 1:M_k.

The created polyline, after being simplified, undergoes verification.

Verification of results

Piątkowski (1969) defined the norm of the accuracy of acquiring objects with character of the curve, calling her the primitive generalization. The original generalisation idea consists in use of line segments (chords) in place of corresponding curvilinear sections of a linear object. Piątkowski determined, empirically, the length of the ordinate - e - between the chord and the arc of the curve, called the rise. Length of the rise depends on the scale of the map as follows:

$e \leq 0{,}3\ M_0$ [mm], (24)

where: M_0 is the source map scale denominator (the map, from which the data is drawn).

The scale denominator M_0, in every simplification process, satisfies the condition:

$e_k \leq 0{,}3\ M_k$,[mm], (25)

where: Mk – the target scale denominator.

For practical reasons of the simplification process, the right hand side of the equation (17) can be replaced by the condition of equality in relation (24), which allows:

- verification of the newly arranged points of the curve with use of the Piątkowski condition (25).
- unambiguous test of the condition (25) for the rises from the deleted points of the original polyline to the sides of the simplified polyline,
- the evaluation of precision of the data obtained in 1:M_k scale.

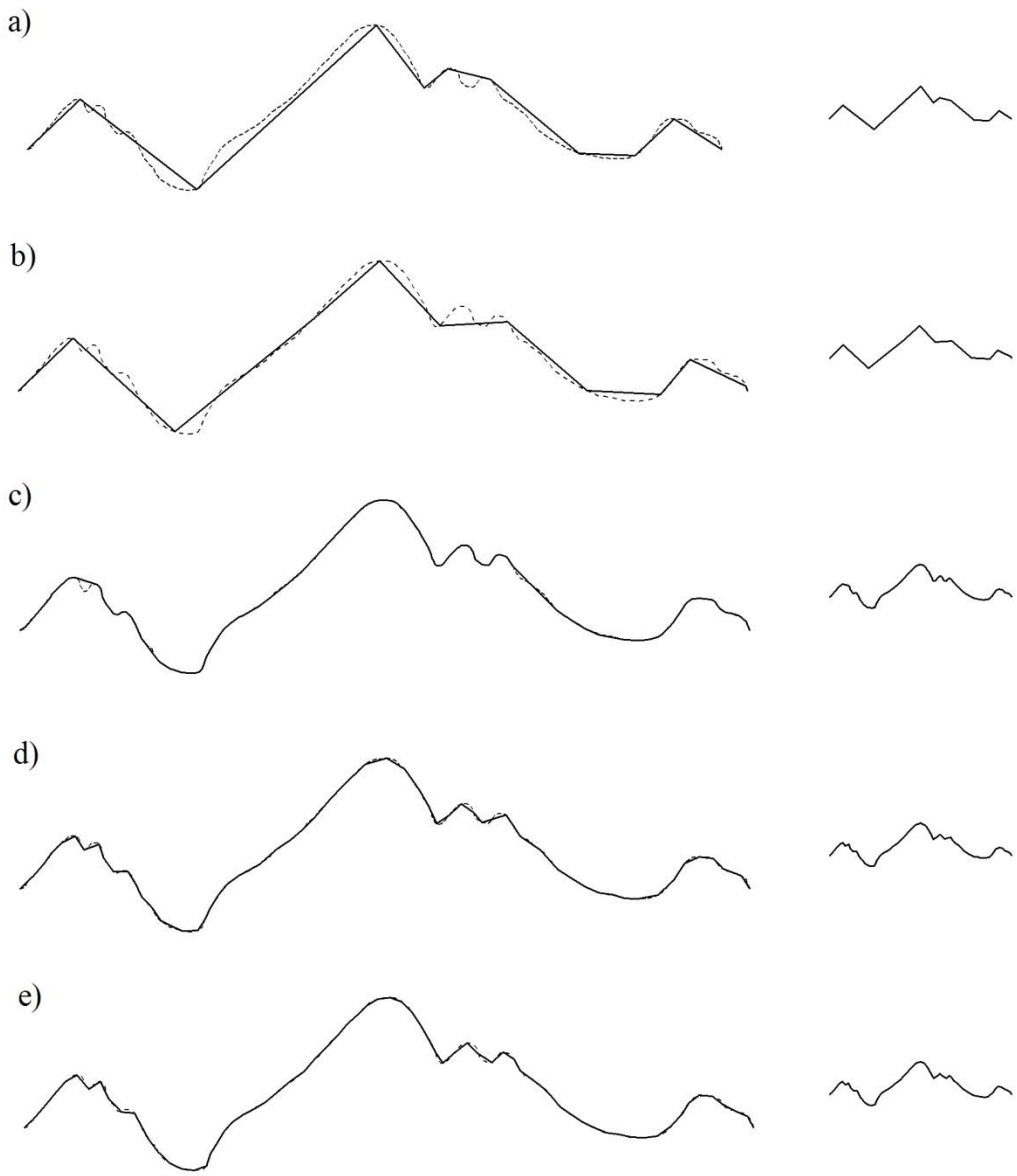

Figure 7. Example of applying the described algorithm on the single part: a) original, b) 1 : 5000, c) 1 : 10 000, d) 1 : 25 000, e) 1: 50 000 (for b, c, d and e below visualisation in the scale)

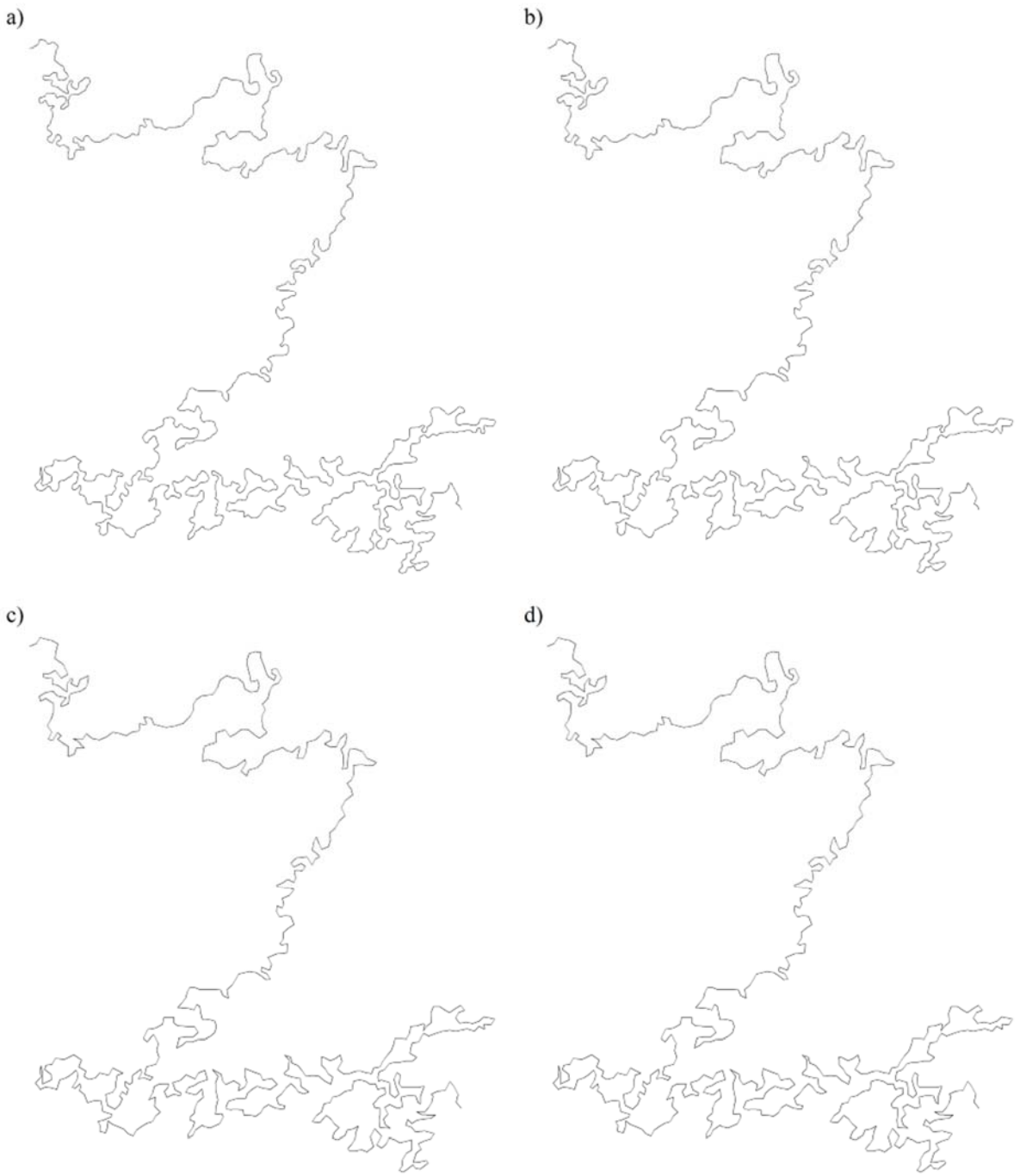

Figure 8. Example of the symplification on large dataset (mor then 10000 vertex): b) 1 : 5000, c) 1 : 10 000, d) 1 : 25 000, e) 1: 50 000 (the data becomes from remote sensing for VHR)

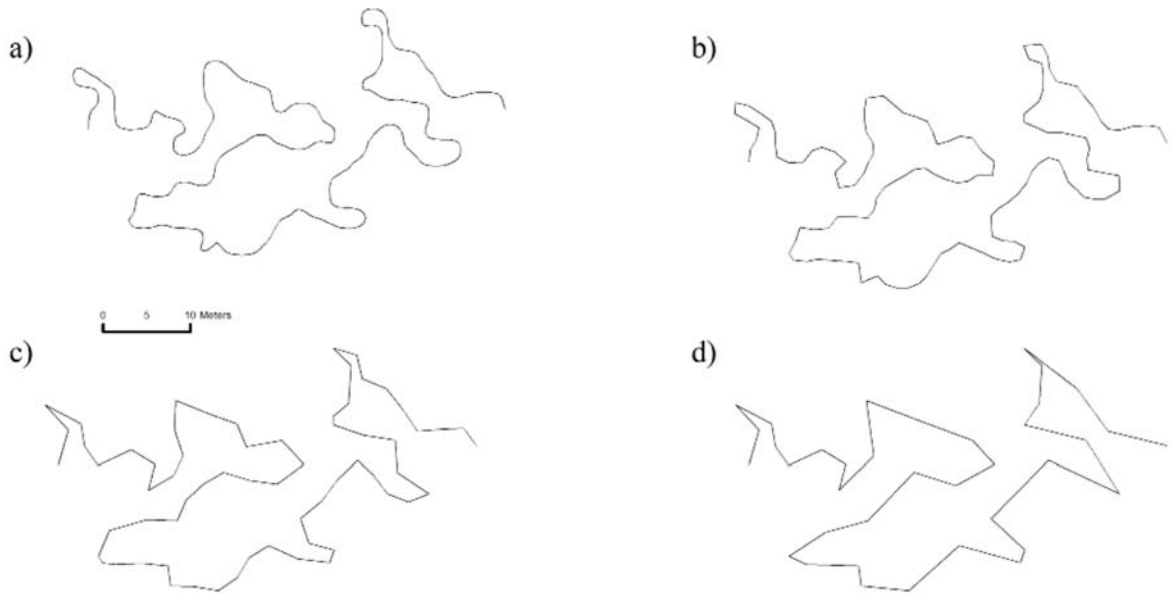

Figure 9. Example of symplification few parts from figure 8: b) 1 : 5000, c) 1 : 10 000, d) 1 : 25 000, e) 1: 50 000 (the data becomes from remote sensing for VHR)

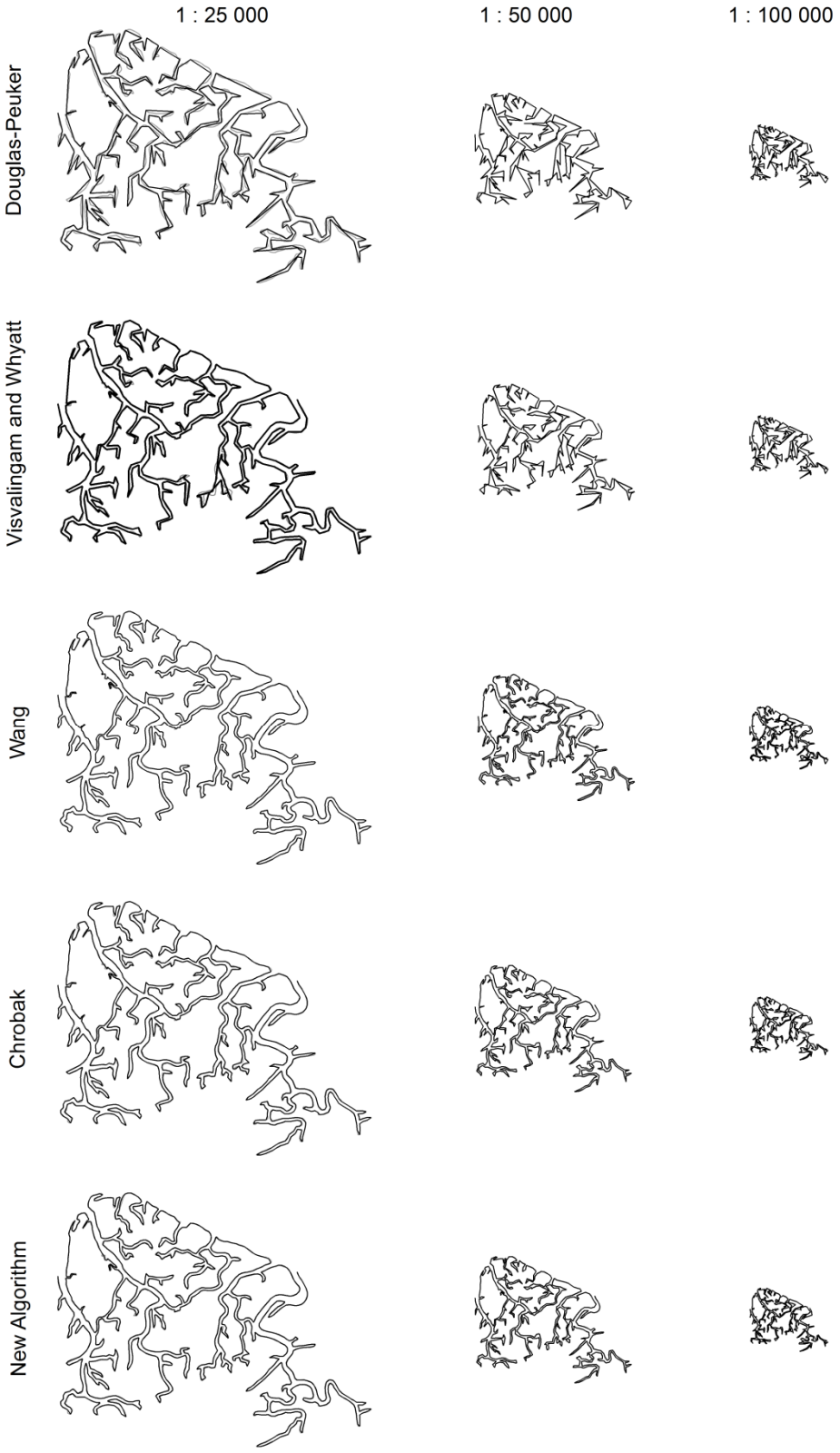

Figure 10. An example of the simplification of a coastline with use of the other algorithms for chosen scales (value of threshold parameter for other algorithms: Douglas and Wang 15, 30, 60 and Visvaligam 50, 250, 1250)

The results of the polylines simplified with Wang algorithm do not exceed the acceptable length of rise up to the 1:10,000 scale and are similar to these obtained in the simplification process using the Chrobak algorithm and the new method. In the 1:25,000 and 1:50,000 scales, the differences in the rise lengths increase significantly, which results from the influence of the arrangement of the points on the original polyline on the size of the (increasing) scale difference, which has been neglected in the simplification algorithms used before.

Table 1. Lengths L_i of the polylines simplified from figure 7 with the compared algorithms and the differences between them and the original polyline – L_0

Algorithm	**Lengths and their differences**	**Scales**					
		1:1000	**1:2000**	**1:5000**	**1:10000**	**1:25000**	**1:50000**
Douglas - Peückera	L_1	390.121	387.045	381.175	367.019	345.856	320.62
	ΔL_1	-3.433	-6.509	-12.379	-26.535	-47.698	-72.934
Visvalingham – Whyatt	L_2	393.171	390.869	379.408	361.306	300.463	284
	ΔL_2	-0.383	-2.685	-14.146	-32.248	-93.091	-109.554
Wang	L_3	393.551	392.928	389.951	381.65	364.037	352.439
	ΔL_3	-0.003	-0.626	-3.603	-11.904	-29.517	-41.115
Chrobak	L_4	393.554	393.262	390.304	385.391	372.131	361.375
	ΔL_4	0	-0.292	-3.25	-8.163	-21.423	-32.179
The new algorithm	L_5	393.554	393.554	391.209	385.651	369.773	361.791
	ΔL_5	0	0	-2.345	-7.903	-23.781	-31.763

The differences $-\Delta L_i$ of the length between the original polylines and these simplified using the Doulas-Peücker and Visvalingham-Whatt algorithms are similar (Table 1). In case of the three other algorithms, the differences $-\Delta L_i$ are significantly smaller. The lengths of the polylines simplified with Wang algorithm show no important difference to the lengths of the polylines simplified with Chrobak algorithm and the new method. The differences are more significant in smaller scales (1:25,000 and 1:50,000), what proves that it is crucial to arrange the curve vertices according to the scale. The great advantage of the new method is the extended range of scales, what is particularly useful in MRDB.

Conclusions

The new algorithm uses the following norms: Piatkowski norm of the orginal generalisation and the standard of the drawing recognition. It enables verification of the hitherto applied algorithms in a way not dependent on the decision of the user.

The conversion of a polyline into a curve (lines of at least the class C_2) allows arrangement of the points on the original curve depending on the target map scale and not source map scale, as practised before.

The presented method can be implemented to any database of MRDB type under the assumption that the high-detail data will constitute the source data, whereas the results will be data for the purpose of visualization at low level of detail.

The Brassel-Weibel (Brassel and Weibel 1988) model for cartographic generalisation distinguishes five stages of processing:

a) structure recognition,

b) process recognition,

c) process modelling (by proper selection of parameters),

d) process execution,

e) verification and visualisation of results.

The lack of independent parameters in the digital cartographic generalisation made it impossible to carry out the last stage of the Brassel-Weibel model: verification. However, the new algorithm enables us to implement it. The standard of the drawing recognition and deterministic method implemented in the simplification process with this particular algorithm gives the unambiguous result as well as the opportunity for a measurable verification.

The simplification process with use of the new algorithm fulfils conditions of a method, because the value of its parameter does not depend on the user, and the result is congruent with accuracy of the applied norms.

References

De Boor, C., 1978. *A Practical Guide to Splines*. Berlin: Springer-Verlag.

Brassel, K.E. and Weibel, R., 1988. A Review and Conceptual Framework of Automated Map Generalization. *International Journal of Geographical Information Science*, 2 (3), 229–244.

Chrobak, T., 2000. A Numerical Method for Generalizing the Linear Elements of Large-Scale Maps, Based on the Example of Rivers. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 37 (1), 49–56.

Chrobak, T., 2010. The role of least image dimensions in generalized of object in spatial databases. *Geodesy and Cartography*, 59 (2), 99–120.

Douglas, D.H. and Peucker, T.K., 1973. Algorithms for the Reduction of the Number of Points Required to Represent a Digitised Line or its Caricature. *Cartographica: The*

International Journal for Geographic Information and Geovisualization, 10 (2), 112–122.

Dyken, C., Dæhlen, M., and Sevaldrud, T., 2009. Simultaneous curve simplification. *Journal of Geographical Systems*, 11 (3), 273–289.

Lang, T., 1969. Rules for robot draughtsmen. *The Geographical Magazine*, 42, 50–51.

Li, Z., 1993. Some observations on the issue of line generalisation. *Cartographic Journal, The*, 30 (1), 68–71.

Li, Z., 2007. *Algorithmic Foundation of Multi-Scale Spatial Representation*. London: CRC Press.

Longley, P.A. *et al.*, 2006. *GIS, Teoria i praktyka*. Warszawa, Polska: Wydawnictwo Naukowe PWN.

Mackaness, W.A. and Ruas, A., 2007. Evaluation in the Map Generalisation Process. *In*: W.A. Mackaness, A. Ruas, and L.T. Sarjakoski, eds. *Generalisation of Geographic Information: Cartographic Modelling and Applications*. Amsterdam, 89–111.

McMaster, R.B., 1991. Conceptual frameworks for geographical knowledge. *In*: B.P. Buttenfield and R.B. Mcmaster, eds. *Map Generalization: Making Rules for Knowledge Representation*. London: Longman Scientific & Technical, 21–39.

McMaster, R.B. and Monmonier, M., 1989. A conceptual framework for quantitative and qualitative raster-mode generalization. *In*: *Proceedings of GIS/LIS'89, Orlando, 26-30 November*. Bethesda, MD: American Congress on Surveying and Mapping, 390–403.

Molenaar, M., 1998. *An Introduction to the Theory of Spatial Object Modelling for GIS*. 1st ed. London: Taylor & Francis.

Perkal, J., 1966. An attempt at objective generalization. *In*: J.D. Nystuen, ed. *Discussion Papers of the Michigan Interuniversity Community of Mathematical Geographers. Discussion Paper 10.* MI: Ann Arbor: Department of Geography, University of Michigan.

Pitkowski, F., 1969. *Kartografia: redakcja map i reprodukcja kartograficzna*. Warszawa: Wydawnictwo Naukowe PWN.

Saliszczew, K.A., 1998. *Cartography.* 2nd ed. Warszawa: Wydawnictwo Naukowe PWN.

Sarjakoski, L.T., 2007. Conceptual Models of Generalisation and Multiple Representation. *In*: W.A. Mackaness, A. Ruas, and L.T. Sarjakoski, eds. *Generalisation of Geographic Information: Cartographic Modelling and Applications*. Amsterdam: Elsevier, 11–35.

Shea, K.S. and McMaster, R.B., 1989. Cartographic generalization in a digital environment: When and how to generalize. *In*: *Proceedings Auto-Carto 9, Ninth International Symposium on Computer-Assisted Cartography, Baltimore, 2-7 April*. Baltimore, 56–67.

Visvalingam, M. and Whyatt, J.D., 1993. Line generalisation by repeated elimination of points. *The Cartographic Journal*, 30 (1), 46–51.

Wang, Z., 1996. Manual versus Automated Line Generalization. *In*: *Proceedings of GIS/LIS '96, Denver, 19-21 November*. Bethesda, MD: American Society for Photogrammetry and Remote Sensing, 94–106.