Proceedings of the International MultiConference of Engineers and Computer Scientists 2009 Vol I
IMECS 2009, March 18 - 20, 2009, Hong Kong

On Realistic Line Simplification under Area Measure

Shervin Daneshpajouh* Alireza Zarei† Mohammad Ghodsi‡

Abstract—**In this paper, we consider the well-known 2D line simplification problem under area measure. Primarily, we propose a unified definition for the area measure that can be used on a general path of n vertices. We present an $O(n^3)$ optimal simplification algorithm on general paths under unified area measure based on Imai and Iri's approach. Next, to improve the time complexity, we describe a realistic situation in which the path lies inside a bounded region. For such a realistic input path, we propose an ϵ-approximate algorithm of $O(\frac{n^2}{\epsilon})$ time complexity to find the simplification. We further define a variant of the area measure that overcomes the pitfalls of the common area measure arisen in degenerated cases. Our optimal or approximation algorithms can employ this measure at the same time and space complexities. Although, the area is the first natural simplification measure, to the best of our knowledge, the results presented here are the first sub-cubic simplification algorithms on this measure for general paths.** ***Keywords: Computational geometry, line simplification, line generalization, area measure, approximation algorithm.***

1 Introduction

Line simplification, also referred to as line generalization in some literatures, is a basic problem in imaging, cartographic, computational geometry and geographic information systems (GIS). In this problem, there is a sequence of input points defining a path $P = \langle p_1, p_2, \ldots, p_n\rangle$ and we are asked to approximate this path by another path $Q = \langle q_1 = p_1, q_2, \ldots, q_k = p_n\rangle$ of smaller number of vertices. The difference between P and Q is measured using different metrics.

There are two main versions of this problem. In the *restricted* version, the vertices q_i of Q must be a subsequence of the vertices of P. In the *unrestricted* version, this restriction does not exist. Some results on the unrestricted version can be found in [1, 2, 3, 4].

In this paper, we only consider the restricted version. For this problem, two optimization goals have been set: (1) min-k, where an error threshold δ is given and the goal is to find the simplification with a minimum number of vertices and error of at most δ, and (2) min-δ, where for a given number k, the goal is to find k vertices (at most) with minimum simplification error. The min-δ version can be solved by a simple binary search from the results of min-k [5]. Therefore, we focus on the min-k problem in this paper.

The error of a simplification Q under a measure m, denoted by $error_m(Q)$, is either defined to be $\max_{i=1}^{k-1} error_m(q_iq_{i+1})$ or $\sum_{i=1}^{k-1} error_m(q_iq_{i+1})$. Assuming that q_iq_{i+1} is the simplification of sub-path $P(s,t) = \langle p_s = q_i, p_{s+1}, \ldots, p_t = q_{i+1}\rangle$, $error_m(q_iq_{i+1})$ is the associated error of approximating $P(s,t)$ by link q_iq_{i+1} under measure m. The main simplification metrics are retained length, angular change, perpendicular distance, Fréchet distance, and areal displacement [6]. A survey and comparison of these metrics can be found in [7, 8].

Related work. The oldest and most popular approach for line simplification is Douglas-Peucker algorithm [9]. A basic implementation of this algorithm for orthogonal distance runs in $O(n^2)$ time. Other implementations improved the running time to $O(n \log n)$ [10] and to $O(n \log^* n)$ [11]. However, this algorithm is a heuristic without any guarantee about the quality of the resulting approximation.

The first general and optimal algorithm was proposed by Imai and Iri[5]. They modelled the problem by a directed acyclic graph and showed that solving the shortest path on this graph is equivalent to the optimal simplification. Moreover, they showed how this graph can be constructed for orthogonal distance measure in $O(n^2 \log n)$ time. This running time was improved to quadratic or near quadratic in [12, 13], and to $O(n^{4/3})$ in [14] for L_1 and uniform metrics. Finally, a near linear time approximation algorithm was proposed in [15] for L_2 orthogonal metric distance.

The line simplification under the Fréchet distance was

*Sharif University of Technology, Computer Engineering Department, P.O. Box 11365-9517, Tehran, Iran. Email: daneshpajouh@ce.sharif.edu
†Sharif University of Technology, Computer Engineering Department, P.O. Box 11365-9517, Tehran, Iran. Email: zarei@mehr.sharif.edu
‡Sharif University of Technology, Computer Engineering Department, P.O. Box 11365-9517 & School of Computer Science, Institute for Research in Fundamental Sciences (IPM), P.O. Box: 19395-5746, Tehran, Iran. Email: ghodsi@sharif.edu

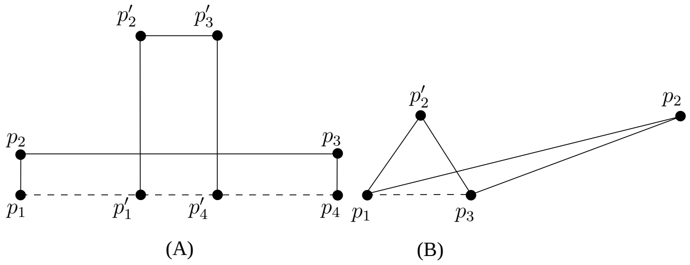

Figure 1: Area measure pitfalls.

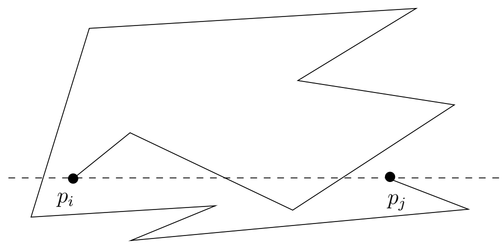

Figure 2: A complex sub-path simplified by p_ip_j.

first studied in [16]. For this metric, the optimal solution can be obtained using the results of [5, 17].

Cromley and Campbell [18] modelled the line simplification problem as a set of linear programming constraints which can be solved using conventional approaches. There are various results for line simplification under the area measure [19, 20, 21, 22, 23, 24, 25]. It was first studied by McMaster [19, 20]. Then, Visvalingam and Whyatt [21] presented a linear simplification algorithm based on the *effective area.* This algorithm uses the effective area as a heuristic to eliminate vertices. However, there is no estimation about the quality of the result of this method.

Moreover, Veregin [22] categorized different metrics and compared the simplification results under orthogonal distance and area measure. Aronov *et al.* [24] studied the min-k simplification under L_1 and L_2 metrics. They presented an approximation algorithm of $O(kn^4\epsilon^{-4}\log^4(n+\epsilon^{-1}))$ time complexity and $1+\epsilon$ error. Bose *et al.* [25] studied the area measures in three categories: Sum-area , Max-area and Diff-area. They proposed simplification algorithms for these three measures on x-monotone paths. For the first model, they presented a polynomial time algorithm for min-k simplification. They showed that the other models are NP-hard and therefore presented approximation algorithms for these measures. The Sum-area model relates to our objective measure and will be described in detail later.

Motivation. Assume that a sub-path $P(i,j)$ has been simplified by the link p_ip_j. It is natural to use the area of the region defined by p_ip_j and $P(i,j)$ as the error of this simplification. Unfortunately, computing this area and running an optimal algorithm for this measure is costly. On the other hand, the orthogonal distance measures like Hausdorff or uniform distance measures can be computed efficiently and have faster algorithms. These metrics define a tolerance zone around the simplification link in which the original path resides. But, there is no estimation for the area contained between the original path and its simplification link.

An optimal algorithm under area measure was described in [22] which computes the error of all simplifications built on all possible combinations of the vertices of the path. However, this algorithm is exponential and is not useful in practice.

On the other hand, approximation or heuristic algorithms, like the method presented by Visvalingam and Whyatt [21], are not efficient in narrow applications. The problem with these methods is that there is no guarantee on the deviation of their results from the optimal solution.

Bose *et al.* [25] presented algorithms for line simplification under area measure only for x-monotone paths. In many applications like map rendering, paths are not x-monotone. Therefore, it is interesting to study the problem in general cases.

Moreover, while the area measure is a good error metric, it does not work well in all situations. Assume that we want to respectively simplify the paths $\langle p_1, p_2, p_3, p_4\rangle$ and $\langle p_1', p_2', p_3', p_4'\rangle$ of Figure 1.A by links p_1p_4 and $p_1'p_4'$. Also, assume that the area of the rectangles $p_1p_2p_3p_4$ and $p_1'p_2'p_3'p_4'$ are equal. Then the error of these simplifications will be equal while it seems that the first simplification is more acceptable. So, the area measure may fail in some situations. To overcome this weakness, the so-called uniform distortion measure has been proposed for which the error of a link is defined to be the area divided by the link length [22]. Using this measure, the error of the link $p_1'p_4'$ will be greater than the error of the link p_1p_4. This resolves the problem in these situations. Also, as seen in Figure 1.B, in some situations this measure will also fail. In this figure, the error of the link p_1p_2 for paths $\langle p_1, p_2, p_3\rangle$ and $\langle p_1, p_2', p_3\rangle$ under both area and area distortion measures, are equal. But, it seems that this simplification for path $\langle p_1, p_2, p_3\rangle$ has a larger error. In this paper, we present a new area dependent metric that avoids such pitfalls.

Our results. Trying to work on the area-based line simplification, we faced different definitions for this measure none of which was applicable on a general path. Therefore, we first describe a unified definition for the area measure that can be applied on general paths. Further, for this unified area measure, we employ Imai and

ISBN: 978-988-17012-2-0

Iri's general approach and propose an optimal $O(n^3)$ simplification algorithm in which the simplification error is $\max_{i=1}^{k-1} error_m(q_i q_{i+1})$.

The time complexity of this algorithm is too high to be used in practical applications. We propose a near quadratic approximation algorithm that approximately finds the optimal simplification under the unified area measure. Precisely, the running time of our algorithm is $O(\frac{n^2}{\epsilon})$ and the resulting simplification is a ϵL^2-approximation of the optimal simplification where L is the length of the longest shortcut of the simplification.

Assuming that in a realistic application the path entirely lies within a given region of bounded widths from which the points of the path are selected, we conclude that the length of the longest link, in a simplification of a path in such scenes, is a constant value. Therefore, our near quadratic time algorithm can obtain an arbitrarily small approximation simplification of a path in such realistic scenes.

As discussed in Motivation, sometimes the area measure produces *bad* approximations. To prevent such cases, we define a new area measure that depends on the length of the original path as well as the length of the simplification link.

The unified area measure is described in Section 2. The optimal simplification algorithm is presented in Section 3. The approximation algorithm is presented in Section 4 and the redefined measure is described in Section 5. Section 6 contains the conclusion.

2 The Area Measure: Revisited

Assume that we have a sub-path $P(i,j) = \langle p_i, p_{i+1}, \ldots, p_j \rangle$ simplified by the link $p_i p_j$. The error of this simplification under the area measure, denoted by $error_{area}(p_i p_j)$ is defined to be the area of the region enclosed by $P(i,j)$ and $p_i p_j$. In general, the sub-path may intersect itself or $p_i p_j$. The enclosed region may be too complex to identify and compute its area. An example of such complex paths is shown in Figure 2. Therefore, we need a definition that covers all paths.

We first distinguish the areas lying below and above a link. Hence, we have two values defining the error of a link $p_i p_j$: the left area and the right area which are respectively denoted by $Area_l(p_i p_j)$ and $Area_r(p_i p_j)$ (If we are at point p_i and looking toward point p_j, some parts of the sub-path lie on our left hand and the other parts lie on the right). As defined below, these values can be either positive or negative. Then, the error of a link $p_i p_j$ is defined as $error_{area}(p_i p_j) = |Area_l(p_i p_j)| + |Area_r(p_i p_j)|$ where $|x|$ is the absolute value of x.

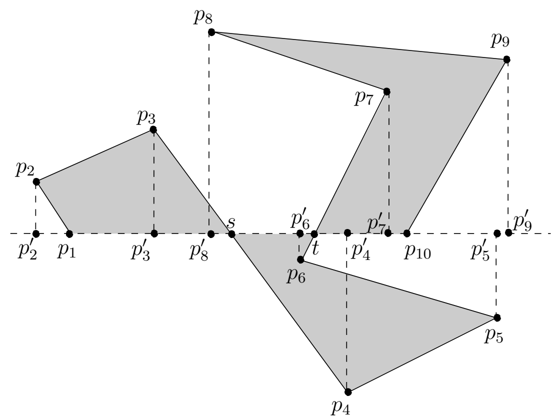

Figure 3: The unified area measure.

Values of $Area_l(p_i p_j)$ and $Area_r(p_i p_j)$ are defined in terms of the areas contained between edges of $P(i,j)$ and $p_i p_j$. Assume that $p_s p_t$ is an edge of $P(i,j)$ where $i \le s < t \le j$. If $p_s p_t$ intersects the supporting line of $p_i p_j$ (the line that includes $p_i p_j$), each part of this edge is handled separately. So, without loss of generality, we assume that $p_s p_t$ lies on the left side of the supporting line of $p_i p_j$. Then, the area between $p_i p_j$ and $p_s p_t$ is equal to the area of the trapezoid $p'_s p_s p_t p'_t$ where p'_s and p'_t are respectively the orthogonal projection of points p_s and p_t on the supporting line of $p_i p_j$. The sign of this area is positive if $\overrightarrow{p_i p_j}$ and $\overrightarrow{p'_s p'_t}$ have the same directions. Otherwise, it is considered as a negative area. Computing these areas for all edges of $P(i,j)$, the values of $Area_l(p_i p_j)$ and $Area_r(p_i p_j)$ are the sum of the corresponding areas of these edges on left or right of $p_i p_j$.

As an example, assume that $p_1 p_{10}$ is the simplification of the sub-path $P(1,10)$ in Figure 3. The area of a polygon $p_i p_{i+1} \ldots p_j$ is denoted by $A(p_i p_{i+1} \ldots p_j)$. According to our definition,

$$\begin{aligned}
Area_l(p_1 p_{10}) &= -A(p_1 p_2 p'_2) + A(p'_2 p_2 p_3 p'_3) + \\
&\quad A(p'_3 p_3 s) + A(t p_7 p'_7) - \\
&\quad A(p'_7 p_7 p_8 p'_8) + A(p'_8 p_8 p_9 p'_9) \\
&\quad -A(p'_9 p_9 p_{10}), \text{ and} \\
Area_r(p_1 p_{10}) &= A(s p_4 p'_4) + A(p'_4 p_4 p_5 p'_5) \\
&\quad -A(p'_5 p_5 p_6 p'_6) + A(p_6 p'_6 t)
\end{aligned}$$

It is simple to verify that this computation is equal to the area of the gray regions in Figure 3.

3 Optimal Algorithm

An efficient general algorithm for restricted, min-k version of line simplification algorithm has been proposed by Imai and Iri [5]. We plug our error function into this algorithm and solve the problem optimally.

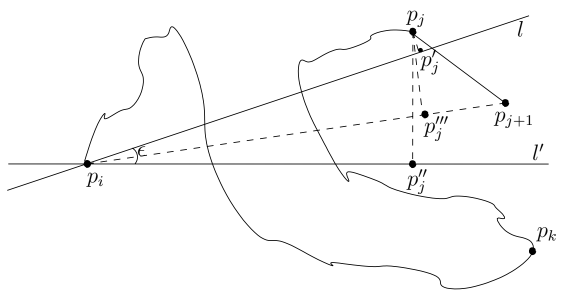

Figure 4: Approximating the error of a link.

The definition of the area measure presented in Section 2 can be used as a unified and general definition and applied on any path. First, we compute $error_{area}(p_ip_j)$ by a linear trace on the sub-path $P(i, j)$ in $O(j - i)$ time. There are $O(n^2)$ possible links for which the unified error must be computed. Consequently, we can do this computation for all p_ip_j links in $O(n^3)$. We build directed acyclic graph G over the vertices of path $P = p_0, p_1, ..., p_n$. and solve the min-k problem as follows:

All edges whose weight (the error of the corresponding link which is $error_{area}(p_ip_j)$) are greater than the given δ are removed from the DAG. Weights of the remaining edges are set to 1. Running a shortest path algorithm from p_1 to p_n returns the optimal min-k simplification. Therefore,

Theorem 1 . *The optimal min-k simplification under the unified area measure can be computed in $O(n^3)$ time and $O(n^2)$ space complexities.*

4 Approximation Algorithm

The time complexity of the optimal $O(n^3)$ algorithm is too high to be used in practical applications. In this section we propose a near quadratic time algorithm to compute the simplification. However, the resulting simplification is not optimal.

The idea of this approximation algorithm is to use the information resulted in computing $error_{area}(p_ip_j)$ to compute $error_{area}(p_ip_{j+1})$ efficiently. This is done by computing and maintaining the error of the current sub-path for a set of canonical lines drawn from the start vertex of the path (here p_i). For the next point, p_{j+1}, we determine the two canonical lines where p_{j+1} lies between them (from now on, we call these two lines l and l'). We approximate the error of p_ip_{j+1} by the errors of these two lines.

Assume that for a sub-path $P(i, j)$ we have the exact value of $Area_l(p_ip'_j)$, $Area_r(p_ip'_j)$, $Area_l(p_ip''_j)$ and $Area_r(p_ip''_j)$ where p'_j and p''_j are respectively the orthogonal projections of p_j on the lines l and l' drawn from p_i (See Figure 4). For the next point p_{j+1}, we use $Area_l(p_ip'_j) + Area_r(p_ip''_j) + S \times A(p_jp'''_jp_{j+1})$ as an approximation for $error_{area}(p_ip_{j+1})$ where l lies on the left of p_ip_{j+1}, l' lies on the right of p_ip_{j+1}, p'''_j is the orthogonal projection of p_j on the supporting line of p_ip_{j+1} and $S \in \{+1, -1\}$ is the sign of the area of the trapezoid defined by p_jp_{j+1} on the supporting line of p_ip_{j+1} (In Figure 4, we have $S = +1$). We denote this approximated value by $error^*_{area}(p_ip_{j+1})$.

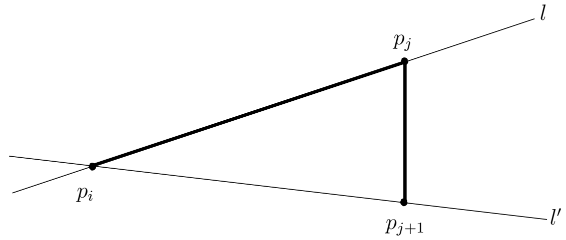

Figure 5: A tight example of the approximation of $error_{area}(p_ip_{j+1})$.

Lemma 1 . *Having the above conditions, we have,*

$$\begin{aligned} error_{area}(p_ip_{j+1}) - \frac{\epsilon|p_ip_k|^2}{2} &\leq error^*_{area}(p_ip_{j+1}) \\ &\textit{and} \\ error^*_{area}(p_ip_{j+1}) &\leq error_{area}(p_ip_{j+1}) \end{aligned}$$

where ϵ is the angle between l and l' containing the point p_{j+1} and p_k is the farthest point of $P(i, j)$ from p_i.

Proof. Assuming that ϵ is small enough, we can simply conclude that $Area_l(p_ip'_j)$ is smaller than $Area_l(p_ip_{j+1})$ and $Area_r(p_ip''_j) \leq Area_r(p_ip_{j+1})$.

This difference is related to the area that lies between l and l'. According to the definition of p_k; this area is at most $\frac{\epsilon|p_ip_k|^2}{2}$ when ϵ is arbitrarily small. Figure 5 shows a tight example. □

Thus, if we have these canonical lines for the small value of ϵ, we can approximate the error of the next point in constant time. However, we need to update the left and right areas of these canonical lines against the newly received vertex to be able to approximate the error of the next point.

Lemma 2 . *There is a $O(\frac{2\pi}{\epsilon}n)$ time algorithm that can approximately compute the error of all links p_1p_i where $1 < i \leq n$.*

Proof. We have $\frac{2\pi}{\epsilon}$ canonical lines from p_1 and on receiving a new point, values of $Area_l$ and $Area_r$ are updated for these lines. Then the approximated error of the new link is computed in constant time. □

For any vertex p_i we can apply the above method. Then, we can find the approximated error of all links $p_i p_j$ in $O(\frac{2\pi}{\epsilon} n^2)$ time. As mentioned before, in a realistic scene we are working in a bounded region. Then the distance between any two points is smaller than a constant value. Therefore, we can omit the $|p_i p_k|$ in Lemma 1. Combining these results,

Theorem 2 . *There is a $O(\frac{n^2}{\epsilon})$ time algorithm that can be used to find an approximation simplification under the unified area measure. The error of the resulting simplification differs from the error of the optimal simplification in $O(\epsilon)$ in a realistic scene.*

If we use these errors as the weight of the DAG in Imai and Iri algorithm, we can approximately compute the simplification under the unified area measure.

5 Improved Unified Area Measure

As described in Figure 1, in some cases, the area measure cannot produce good simplifications. Specifically, the area measure does not consider the length of the sub-path simplified by a link as well as the length of the link itself. As described before, the uniform areal distortion measure has been defined to consider lengths of the simplified links. But, there is no formal method which considers both length of the sub-path and length of the link as well as the enclosed area between them.

Here, we define a new measure, named *relative unified area* which considers both these lengths as well as the unified area. This measure is defined as

$$\text{relative unified area} = \text{unified area} * \frac{\text{the sub-path length}}{\text{the link length}}.$$

This measure can reasonably handle both cases shown in Figure 1. Moreover, this measure can be computed in the same time and space complexities as we did for the unified area measure. Therefore, we can interchangeably use the unified area or the relative unified area measures in the presented algorithms to simplify a general path.

6 Conclusion

In this paper, we considered the well-known line simplification problem under the area measure. Several heuristics or optimal algorithms were proposed for this problem that has many applications in different areas. The previously proposed optimal algorithms either are too costly to be used in real applications or work only on the special case of x-monotone paths. Therefore, heuristic and non-optimal solutions are always used in real applications.

We first proposed a unified definition for area measure that can be used on any paths. Then, we described an algorithm that guarantees the production of optimal solution in $O(n^3)$. Furthermore, we suggested a near quadratic approximation algorithm that can be used for simplifying a general 2D path under this unified area measure. In a realistic application, the length of the outer boundary is at most L, a constant value. Thus, the approximation factor of the resulting simplification would be ϵ in this case.

Finally, we pointed to some weaknesses of the area and other related measures and proposed a new uniform measure that resolves these weaknesses. For future work, the new measure can be experimentally compared with other measures. Another interesting direction for future studies is simplifying under the area measure for streaming input models . As a blocking issue, the time complexity of the optimal and approximation algorithms is high. Proposing sub-quadratic or near linear approximation algorithms is another open direction in extending this work.

References

[1] Goodrich, M.T., “Efficient piecewise-linear function approximation using the uniform metric,” *Journal of Discrete Computational Geometry*, V14, N1, pp. 445462, 1/05.

[2] Guibas, L.J., Hershberger, J.E., Mitchell, J.S.B., Snoeyink, J.S., “Approximating polygons and subdivisions with minimum link paths,” *International Journal of Computational Geometry & Applications*, V3, pp. 383415, 93.

[3] Hakimi, S.L., Schmeichel, E.F., “Fitting polygonal functions to a set of points in the plane,” *CVGIP: Graph. Models Image Process*, V53, pp. 132136, 91.

[4] Imai, H., Iri, M., “An optimal algorithm for approximating a piecewise linear function,” *Journal of Information Processing*, V9, N3, pp. 159–162, 86.

[5] Imai, H., Iri, M., “Polygonal approximations of a curve-formulations and algorithms,” *Computational Morphology*, North-Holland, pp. 71–86, 88.

[6] Weibel, R., “Generalization of Spatial Data - Principles and Selected Algorithms,” *In: Van Kreveld, M., Nievergelt, J., Roos, Th. and Widmayer, P. (eds.): Algorithmic Foundations of GIS. Lecture Notes in Computer Science (LNCS)*, Berlin: Springer-Verlag, V1340, pp. 99-152, 97.

[7] McMaster, R.B., “Automated Line Generalization,” *Cartographica*, V24, N2, pp. 74-111, 87.

[8] Buzer, L., “Optimal Simplification of Polygonal Chain for Rendering,” *23rd ACM Symposium on Computational Geometry (SoCG)*, pp. 168–174, 07.

[9] Douglas, D.H., Peucker, T.K.: "Algorithms for the reduction of the number of points required to represent a digitized line or its caricature," *Cartographica: The International Journal for Geographic Information and Geovisualization*, V10, N2, pp. 112122, 73.

[10] Hershberger, J., Snoeyink, J.: "An O(n log n) implementation of the Douglas-Peucker algorithm for line simplification," *In Proceeding of 10th ACM Symposium on Computational Geometry*, pp. 383384, 94.

[11] Hershberger, J., Snoeyink, J.: "Cartographic line simplification and polygon CSG formulae in O(n log* n) time," *In Proceeding of 5th International Workshop on Algorithms and Data Structures (WADS)*, LNCS 1272, pp. 93103, 97.

[12] Chan, W.S., Chin, F.: "Approximation of polygonal curves with minimum number of line segments," *In Proceeding of 3rd Annual International Symposium on Algorithms and Computation*, LNCS 650, pp. 378–387, 92.

[13] Melkman, A., ORourke, J.: "On polygonal chain approximation," *In: G.T. Toussaint (ed.), Computational Morphology*, North-Holland, pp. 87–95, 88.

[14] Agarwal, P.K., Varadarajan, K. R.: "Efficient algorithms for approximating polygonal chains," *Discrete Computational Geometry*, V23, pp. 273–291, 00.

[15] Agarwal, P.K., Har-Peled, S., Mustafa, N.H., Wang, Y.: "Near-linear time approximation algorithms for curve simplification," *Algorithmica*, V42, pp. 203–219, 05.

[16] Godau., M.: "A natural metric for curves: Computing the distance for polygonal chains and approximation algorithms," *In Proceeding of 8th Annual Symposium on Theoretical Aspects of Computer Science.(STACS)*, pp. 127–136, 91.

[17] Alt, H., Godau, M.: "Computing the Fréchet distance between two polygonal curves," *International Journal of Computational Geometry and Applications*, V5, pp. 75–91, 95.

[18] Cromley, R. G., CAMPBELL, G. M.: "Noninferior Bandwidth Line Simplification: Algorithm and Structural Analysis," *Geographical Analysis*, V23, N1, pp. 25-38, 91.

[19] McMaster, R.B.: *A Quantitative Analysis of Mathematical Measures in Linear Simplification*, Ph.D. Thesis, Dept. of Geography and Meteorology, University of Kansas, Lawrence, Kansas, 83.

[20] McMaster, R.B.: "The Geometric Properties of Numerical Generalization," *Geographical Analysis*, 19(4) : 330-346 (1987)

[21] Visvalingam, M., Whyatt ,J.D.: "Line Generalisation by Repeated Elimination of Points," *Cartographic Journal*, V30, N1, pp. 46-51, 93.

[22] Veregin, H.: "Line Simplification, Geometric Distortion, and Positional Error," *Cartographica*, V36, N1, pp. 25-39, 99.

[23] Chen, L., Wang, J., Xu, J.: "Asymptotically Optimal and Linear-time Algorithm for Polygonal Curve Simplification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 05.

[24] Aronov, B., Asano, T., Katoh, N., Mehlhorn, K., Tokuyama, T.: "Polyline fitting of planar points under min-sum criteria," *International Journal of Computational Geometry and Applications*, V16, pp. 97-116, 06.

[25] Bose, P., Cabello, S., Cheong, O., Gudmundsson, J., Kreveld, M., Speckmann, B., "Area-Preserving Approximations of Polygonal Paths," *Journal of Discrete Algorithms*, V4, N4, pp. 554-566, 06.