

## Polygonal Approximation of Digital Curves Using a Multi-objective Genetic Algorithm

Hervé Locteau, Romain Raveaux, Sébastien Adam, Yves Lecourtier, Pierre  
Héroux, Eric Trupin

► **To cite this version:**

Hervé Locteau, Romain Raveaux, Sébastien Adam, Yves Lecourtier, Pierre Héroux, et al.. Polygonal Approximation of Digital Curves Using a Multi-objective Genetic Algorithm. IAPR International Workshop on Graphics Recognition, 2005, Honk-Kong, Hong Kong SAR China. pp.300-311, 10.1007/11767978\_27 . hal-00601529

**HAL Id: hal-00601529**

**<https://hal.archives-ouvertes.fr/hal-00601529>**

Submitted on 18 Jun 2011

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Polygonal Approximation of Digital Curves Using a Multi-objective Genetic Algorithm

Herve Locteau, Romain Raveaux, Sebastien Adam, Yves Lecourtier,  
Pierre Heroux, and Eric Trupin

LITIS, Université de Rouen,  
F-76800 Saint-Etienne du Rouvray, France  
Herve.Locteau@univ-rouen.fr

**Abstract.** In this paper, a polygonal approximation approach based on a multi-objective genetic algorithm is proposed. In this method, the optimization/exploration algorithm locates breakpoints on the digital curve by minimizing simultaneously the number of breakpoints and the approximation error. Using such an approach, the algorithm proposes a set of solutions at its end. This set which is called the Pareto Front in the multi objective optimization field contains solutions that represent trade-offs between the two classical quality criteria of polygonal approximation : the Integral Square Error (ISE) and the number of vertices. The user may choose his own solution according to its objective. The proposed approach is evaluated on curves issued from the literature and compared with many classical approaches.

## 1 Introduction

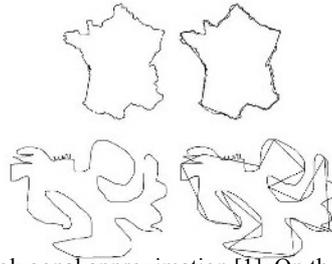
Polygonal approximation of digital planar curves is an important issue in pattern recognition and image processing. It is a classical way to represent, store and process digital curves. For example, its results are frequently used for shape recognition. The problem can be stated as follows: Given a curve  $C$  consisting of  $N$  ordered

points  $C = \{C_i \equiv (x_i, y_i)\}_{i=1}^N$ , the goal is to find a subset  $S$  of  $M$  ordered points  $i$

$S = \{S_i \equiv (x_i, y_i)\}_{i=1}^M$  with  $M \leq N$  and  $S \subseteq C$ . These points constitute the extremities of line segments so that the polygon constructed by directly connecting these line segments best fits the given digital curve. Figure 1, issued from [1], illustrates such a process for two different curves. Many paradigms have been proposed in the literature to solve the problem of polygonal approximation, what leads to a great number of published papers. Among these approaches, one can cite split or split and merge techniques [2][3][4], Hough transform [5], perceptual organization [6], dominant points detection [7][8][9][10][11][12], competitive Hopfield neural networks [13] or dynamic programming [14][15]. Another paradigm has been recently proposed in [16][17][18]. It consists in using Genetic Algorithms in order to find a near-optimal polygonal approximation. In such an approach, as in dynamic programming methods, the polygonal approximation technique is considered as an optimization process and the algorithm automatically selects the best points of the polygon by

minimizing a given criterion. Two kinds of configuration may be distinguished in the published papers. In the first case, the number of vertices to be obtained is fixed and the method uses the concept of genetic evolution to obtain a near-optimal polygon [16][17]. In the second case, a maximal approximation error is fixed and the algorithm minimizes the number of vertices of the polygon [16]. One can note that this kind of approach has also been used in order to approximate curves with circular arcs or ellipses [18][19].

In this paper, we adopt the same paradigm and we propose a new algorithm for polygonal approximation using genetic algorithms. The originality of the described approach is the factorization of the two kinds of configuration mentioned above through the use of a multi-objective optimization process whereas existing approaches minimize a unique criterion as explained before. Such a new viewpoint enables the user of the system to choose a trade-off between different quality criteria, according to its objectives of use of the results (recognition, interpretation, storage...).



**Fig. 1.** Results provided by a polygonal approximation [1]. On the top left, a curve composed of 992 points. On the top right, its polygonal approximation (15 line segments). At the bottom left, a free curve of 3222 points, on the right, its polygonal approximation (16 line segments).

The remainder of the paper is organized as follows. In section 2, an introduction to the multi-objective optimization problem is proposed and our algorithm is detailed. In section 3, the application of this algorithm to the polygonal approximation problem is shown. Section 4 presents the experimentally obtained results, a comparison with classical approaches and a discussion concerning the interests of such an approach. Section 5 summarizes the concluding remarks and proposes some perspectives for this work.

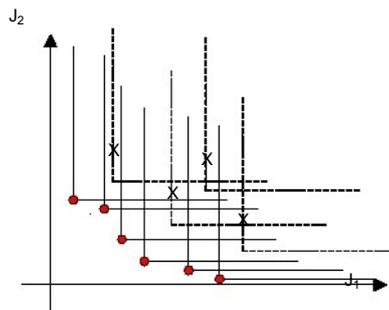
## **2 A Genetic Based Multi-objective Optimization Algorithm**

When an optimisation problem involves more than one objective function (a very frequent context in the document analysis field - one can cite recognition rate/reject rate, precision rate/recall rate, compression / quality), the task of finding one or more optimum solutions is known as multi-objective optimization. Some classical text-books on this subject have been published, e.g. [20]. We just recall here some essential notions in order to introduce the proposed algorithm. The main difference between single and multi-optimization task lies in the requirement of compromises

between the various objectives in the multi-optimization case. Even with only two objectives, if they are conflicting, the improvement of one of them leads to a deterioration of the other one. For example, in the context of polygonal approximation, the decrease of the approximation error always leads to an increase of the vertices number. Two main approaches are used to overcome this problem in the literature. The first one is to combine the different objectives in a single one (the simpler way being to use a linear combination of the various objectives), and then to use one of the well-known techniques of single objective optimization (like gradient based methods, simulated annealing or classical genetic algorithm). In such a case, the compromise between the objectives is a priori determined through the choice of the combination rule. The main critics addressed to this approach are the difficulty to choose a priori the compromise and the fact that some objective points can not be reached. It seems a better idea to postpone this choice after having several candidate solutions at hand. This is the goal of Pareto based method using the notion of dominance between candidate solutions. A solution dominates another one if it is better for all the objectives.

This dominance concept is illustrated on figure 2. On this example, two criteria  $J_1$  and  $J_2$  have to be minimized. The set of non-dominated points that constitutes the Pareto-Front appears as 'O' on the figure, while dominated solutions are drawn as 'X'.

Using such a dominance concept, the objective of the optimization algorithm becomes to determine the Pareto front, that is to say the set of non-dominated points. Among the optimization methods that can be used for such a task, genetic algorithms are well-suited because they work on a population of candidate solutions. They have been extensively used in such a context, with many variants. The most common algorithms are VEGA - Vector Evaluated Genetic Algorithm - [21], MOGA - Multi Objective Genetic Algorithm - approach [22], NSGA - Non-Dominated Sorting Genetic Algorithm - [23], NSGA II [24], PAES - Pareto Archived Evolution Strategy - [25] and SPEA - Strength Pareto Evolutionary Algorithm - [26]. The strategies used in these contributions are different, but the obtained results mainly vary from the convergence speed point of view. A good review of the existing approaches can be found in [27].



**Fig. 2.** Illustration of the Pareto Front concept

The proposed genetic algorithm is elitist and steady-state. This means that (i) it manages two populations and (ii) the replacement strategy of individuals in the populations is not made as a whole, but individual per individual. The two populations are a classical population, composed of evolving individuals and an "archive" population

composed of the current Pareto Front approximation elements. These two populations are mixed during the genetic algorithm. The first population guarantees space exploration while the archive guarantees the exploitation of acquired knowledge and the convergence of the algorithm.

Based on such concepts, our optimization method uses the algorithm #1. In this section, we only describe the optimization algorithm, without applying it to the polygonal approximation problem. The particular application to the polygonal approximation problem is described in the following section. This algorithm has been designed in order to be applied to various problems (see section 5). The design of a new application only consists in the choice of a coding scheme for individuals, in the design of the evaluation method and in the choice of the genetic parameters values.

**Algorithm #1.** The multi objective algorithm in use

*Population Initialization*

*Archive Initialization (selection of the non dominated element in the population)*

**do**

- *Random selection of two individuals  $I_1$  and  $I_2$  in the current population*
- *Crossover between the selected individuals to generate  $I_3$  and  $I_4$*
- *Mutation applied to the generated children  $I_3$  and  $I_4$*
- *Evaluation of children  $I_3$  and  $I_4$*
- *Selection either of the dominant individual  $I_5$  between mutated children (if it exists) or random selection of  $I_5$  between  $I_3$  and  $I_4$*
- *Random selection of an individual in the archive population ( $I_6$ )*
- *Crossover between  $I_5$  and  $I_6$  to generate children  $I_7$  and  $I_8$*
- *Evaluation of children  $I_7$  and  $I_8$*
- *Test for the integration of  $I_7$  and  $I_8$  in the archive*
- *Test for the integration of  $I_7$  and  $I_8$  in the current population*
- *Incrementation of the generation number*

**While** *the maximal generation number has not been reached*

In the current implementation of this algorithm, the coding of an individual is a classical bit set, the crossover is a well-known 2-point crossover, and the mutation is a simple transformation of a gene value by its complementary value. Concerning the replacement strategy, several choices can be made for the integration of a candidate individual in the archive. The simplest is a dominance test between the candidate and the archive elements. The candidate is inserted within the archive if no archive element dominates it. In the same time, archive elements dominated by the candidate are eliminated from the archive. A problem reported in the literature on evolutionary multi-objective optimization is the possible bad exploration of Pareto front: the archive population elements concentrate on only some parts of the front. This difficulty is overcome in our approach by defining a minimal distance between two points

in the objective space. This algorithm has been tested on classical multi-objective problems such as BNH, TNK or OSY [28] problems and the results have shown the quality of the proposed approach.

### 3 A Genetic Based Multi-objective Optimization Algorithm

In order to apply the algorithm presented above to the polygonal approximation problem, an individual has to represent a possible solution to the polygonal approximation problem. That is why an individual is composed of  $N$  genes, where  $N$  is the number of points in the initial curve. A gene is set to '1' if the point is kept as a breakpoint, '0' if it is not. An example of an individual coding is given in figure 3. Each point  $C_i$  of the curve  $S$  corresponds to a bit in the chromosome. In the example of figure 3, the individual is a binary string of 14 genes corresponding to the initial  $C_0 - C_{13}$ . The example polygon is composed of 4 vertices whose extremities are  $C_0$ ,  $C_4$ ,  $C_7$  and  $C_{11}$ . Such a polygonal approximation corresponds to the individual "10001001000100".

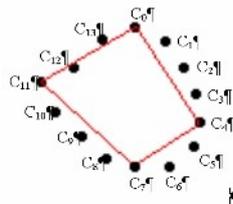


Fig. 3. An example of the coding scheme applied on a simple curve

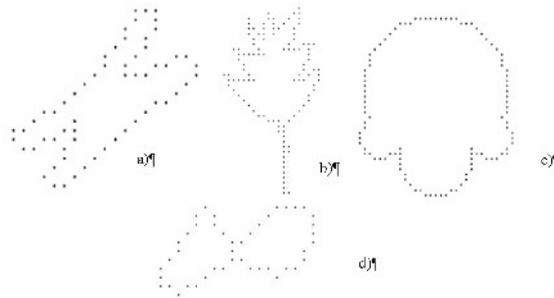
Concerning the computation of the criterion to be optimized, two objectives have been included in the current version. The first one is the Integral Square Error (ISE) and the second one is the vertices number. Such a choice enables to have a trade-off between the precision of the result and the number of line segments, thanks to elements of the Pareto front. One can note that the use of a discrete objective (vertices number) guarantees itself the diversity on the Pareto front, we do not need to specify any minimal distance between any couples of solutions of the Pareto Front.

### 4 Experimental Results and Performance Analysis

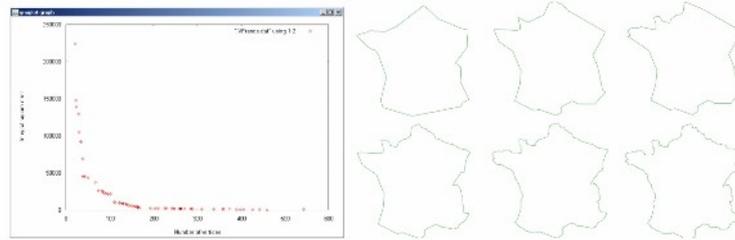
In order to assess the performances of the proposed algorithm, it has been applied to the four digital curves presented in [7]. Fig. 4a is a chromosome shaped curve with 60 points, Fig. 4b is a leaf-shaped curve with 120 points, Fig. 4d is a curve containing four semi-circles with 102 points and Fig. 4c is a figure-eight curve with 45 points. These curves have been broadly used in the literature. Such tests allow to test the performances of the proposed algorithm versus those of published approaches. For each of these curves, the program has been run for 2000 generations, using a

population size of 100 individuals. Such a parameter set involves about 8000 calls to the evaluation method (see the algorithm below). The mutation rate has been fixed to 0.05 and the crossover rate to 0.6. As said before, the output of the presented algorithm is not a single ISE for a number of vertices given a priori. It consists in the whole Pareto front of the optimization problem. That is why the result is a set of couple (ISE - number of vertices). As an example, figure 5 shows the set of couple obtained at the end of one run on a "map of France" curve and the corresponding polygons. Another remark has to be done. Since GA are stochastic, results may be different at independent runs. That is why, in these experiments, we present the best and the worst ISE for each number of vertices obtained after 5 independent runs. Using such a strategy, obtained results are compared for each curve with the results proposed in the literature using the following scheme:

- A table containing the results obtained by several approaches from the literature on the given figure;
- The results (comparable points on the Pareto Front) obtained by the proposed algorithm on the same curve compared with the best ISE found in the literature;
- A figure representing the whole Pareto Front and showing a visual performance comparison between our results and the results issued from the state of the art.



**Fig. 4.** Test curves: (a) chromosome-shaped curve: 60 points; (b) leaf-shaped curve: 120 points; (c) curve with four semi-circles: 102 points and (d) figure-of-eight curve: 45 points



**Fig. 5.** Results obtained on a curve of a map of France: on the left, the obtained Pareto front, on the right, the polygons corresponding to the points labeled (1-6) on the Pareto front

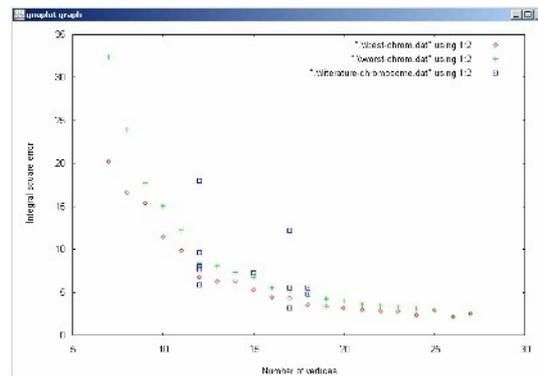
#### 4.1 Results Obtained on the Chromosome-Shaped Curve

**Table 1.** Results obtained with several existing methods on the chromosome-shaped curve

Algorithm	Number of vertices	Integral square error
(Wall and Danielsson, 1984) [4]	17	12.19
(Teh and Chin, 1987) [7]	15	7.20
(Ray and Ray, 1992a) [9]	18	4.81
(Ray and Ray, 1992b) [10]	18	5.56
(Comic, 1997) [11]	12	9.57
(Comic, 1997) [11]	17	5.54
(Perez and Vidal, 1994) [14]	12	5.82
(Perez and Vidal, 1994) [14]	17	3.13
(Huang and Sun, 1999) [17]	12	7.63
(Yin, 1998) [16]	12	7.99
(Marji and Siy, 2003) [12]	12	8.03
(Hornig and Li, 2002) [15]	12	5.82

**Table 2.** Results obtained using 5 runs of the described approach on the chromosome-shaped curve compared with best results found in the literature

Number of vertices	10	11	12	13	14	15	16	17	18	19	20
Min. ISE	11.5	9.8	6.7	6.3	6.3	5.3	4.5	4.3	3.5	3.4	3.2
Max. ISE	15.1	12.2	8.4	8.1	7.3	6.8	5.5	5.0	4.7	4.2	3.9
Best ISE	-	-	5.8	-	-	7.2	-	3.1	4.8	-	-



**Fig. 6.** Comparison between performances obtained using several approaches issued from the literature (•); our best result (o) and our worst result (+) for 5 runs on the chromosome-shaped curve

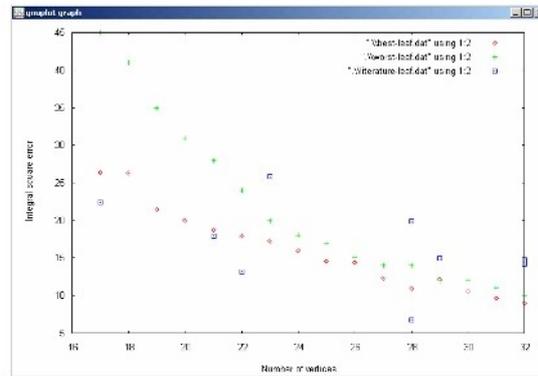
## 4.2 Results Obtained on the Leaf-Shaped Curve

**Table 3.** Results obtained with several existing methods on the leaf-shaped curve

Algorithm	Number of vertices	Integral square error
(Teh and Chin, 1987) [7]	29	14.96
(Ray and Ray, 1992a) [9]	32	14.18
(Ray and Ray, 1992b) [10]	32	14.718
(Cornic, 1997) [11]	23	25.8
(Cornic, 1997) [11]	28	19.88
(Perez and Vidal, 1994) [14]	17	22.42
(Perez and Vidal, 1994) [14]	28	6.80
(Huang and Sun, 1999) [17]	21	17.96
(Marji and Siy, 2003) [12]	22	13.21
(Hornng and Li, 2002) [15]	17	22.42

**Table 4.** Results obtained using 5 runs of the described approach on the leaf-shaped curve compared with best results found in the literature

Number of vertices	17	21	22	23	24	25	26	27	28	29
Min. ISE	26.3	18.8	17.8	17.3	16.0	14.5	14.4	12.4	10.9	12.1
Max. ISE	45.3	28.2	24.2	20.0	18.3	17.2	15.9	15.0	14.5	12.7
Best ISE	22.4	18.0	13.2	25.8	-	-	-	-	6.8	15.0



**Fig. 7.** Comparison between performances obtained using several approaches issued from the literature ( $\diamond$ ); our best result ( $\circ$ ) and our worst result ( $+$ ) for 5 runs on the leaf-shaped curve

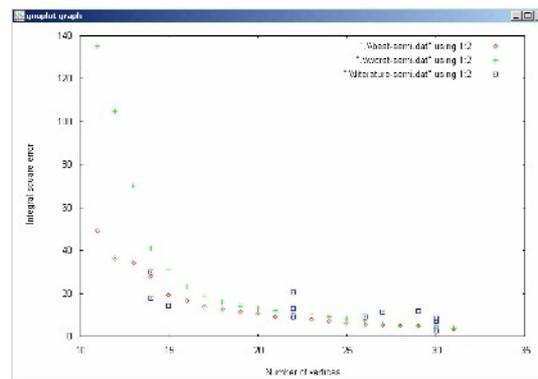
### 4.3 Results Obtained on the Curve with 4 Semi-circles

**Table 5.** Results obtained with several existing methods on the curve with 4 semi-circles

Algorithm	Number of vertices	Integral square error
(Wall and Danielsson, 1984) [4]	30	7.03
(The and Chin, 1987) [7]	22	20.61
(Ray and Ray, 1992a) [9]	27	11.50
(Ray and Ray, 1992b) [10]	29	11.82
(Comic, 1997) [11]	22	13.00
(Comic, 1997) [11]	30	8.38
(Perez and Vidal, 1994) [14]	15	14.40
(Perez and Vidal, 1994) [14]	30	2.64
(Huang and Sun, 1999) [17]	14	17.74
(Huang and Sun, 1999) [17]	22	9.02
(Yin, 1998) [16]	14	29.93
(Yin, 1998) [16]	22	12.91
(Marji and Siy, 2003) [12]	26	9.01
(Hornig and Li, 2002) [15]	15	14.40

**Table 6.** Results obtained using 5 runs of the described approach on the curve with 4 semi-circles compared with best results found in the literature

Number of vertices	14	15	18	20	22	24	26	27	29	30	31
Min. ISE	27.8	19.2	12.7	10.3	8.7	7.0	5.7	5.1	4.8	4.2	3.7
Max. ISE	41.6	23.7	16.4	13.7	11.5	9.5	7.0	6.2	5.6	5.0	4.7
Best ISE	17.7	14.4	-	-	9.0	-	9.0	11.5	11.8	2.6	-



**Fig. 8.** Comparison between performances obtained using several approaches issued from the literature ( ); our best result (o) and our worst result (+) for 5 runs on the curve with 4 semi-circles

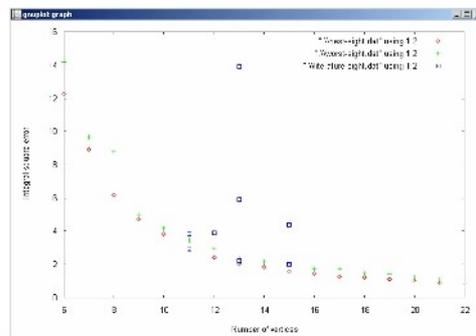
#### 4.4 Results Obtained on the Figure-Eight Curve

**Table 7.** Results obtained with several existing methods on the curve with 4 semi-circles

Algorithm	Number of vertices	Integral square error
(Wall and Danielsson, 1984) [4]	13	13.92
(Teh and Chin, 1987) [7]	13	5.93
(Ray and Ray, 1992a) [9]	15	4.39
(Comic, 1997) [11]	12	3.89
(Perez and Vidal, 1994) [14]	11	2.90
(Yin, 1998) [16]	11	3.83
(Yin, 1998) [16]	13	2.24
(Yin, 1998) [16]	15	2.01
(Hornig and Li, 2002) [15]	11	2.90

**Table 8.** Results obtained using 5 runs of the described approach on the curve with 4 semi-circles compared with best results found in the literature

Number of vertices	8	9	10	11	12	13	14	15	16	17	18
Min. ISE	6.2	4.7	3.8	3.3	2.4	2.0	1.8	1.6	1.4	1.3	1.2
Max. ISE	8.8	5.0	4.1	3.5	2.9	2.4	2.2	2.0	1.8	1.7	1.4
Best ISE	-	-	-	2.9	3.9	2.2	-	2.0	-	-	-



**Fig. 9.** Comparison between performances obtained using several approaches issued from the literature ( ); our best result (o) and our worst result (+) for 5 runs

#### 4.5 Discussion

Many observation can be drawn from the results presented above. Concerning the obtained performances for a given number of vertices, one can see that the proposed approach provides results that are only outperformed by optimal methods [14][15]. These results are logical since GA are known to be near-optimal. On the contrary, the non-optimal or sub-optimal approaches are outperformed by the proposed algorithm.

Another advantage of our approach is the fact that no input parameters concerning the initial curve or the desired results have to be defined a priori. All the tests mentioned above have been led with exactly the same configuration of the program. The most important in this context is that our approach does not need the number of vertices to be obtained as an a priori parameter. Furthermore, the algorithm gives the user a better information on the solved problem as it proposes as output a set of solutions containing a wide range of values for the number of vertices and the corresponding ISE. The user of the system can choose the better solution regarding its application constraints, as shown in Fig. 4. Concerning the complexity, our algorithm has an important computational cost (comparable to Yin's one) but for a pool of solutions. Moreover, this computational cost may be reduced using a parallelization since genetic algorithms are parallel by nature. Another solution to reduce this complexity is to consider an evolution of the genetic parameters (mutation and crossover rates) during the algorithm. This point is actually under consideration.

## 5 Conclusion and Future Works

In this paper, we have proposed a new approach for the polygonal approximation of curves. This approach considers the polygonal approximation as an optimization process. The fundamental difference with existing approaches lies in the fact that we use a multi-objective optimization process while other contributions only optimize a unique objective. One can see several interests in such an approach. As many solutions are proposed, the user may choose the optimal solution regarding its constraints. Another interest is that it is easy to add a new objective. For example, our current work concerns the maximization of the parallel line segments, in order to apply contour matching in a vectorization process. Another future work concerns the integration of the detection of circular arcs in our approximation system, using an approach inspired from [18]. A more global objective is to generalize the principles of multi-objective optimization to the different steps constituting the chain of a document image analysis system. The aim is to build a multi-objective document analysis system which adapts its objective thanks to a dialog with the user.

## References

1. Salotti, M.: An efficient algorithm for the optimal polygonal approximation of digitized curves", PRL 22 (2001) 215-221
2. Ramer, U.: An iterative procedure for the polygonal approximation of plane curves, CGIP 1 (1972) 291-297
3. Pavlidis, T., Horowitz, S.L.: Segmentation of plane curves, IEEE Transaction on Computers 23 (1974) 860-870
4. Wall, K., Danielsson, P.E.: A fast sequential method for polygonal approximation of digitized curves, CVGIP 28 (1984) 220-227
5. Gupta, A., Chaudhury, S., Parthasarathy, G.: A new approach for aggregating edge points into line segments, PR 26 (1993) 1069-1086
6. Hu, J., Yan, H.: Polygonal approximation of digital curves based on the principles of perceptual organization, PR 30 (2002) 701-718

7. Teh, C., Chin, R.T.: On the detection of dominant points on digital curves, IEEE transaction on PAMI 23 (1989) 859-872
8. Ansari, N., Delp, E.J.: On detecting dominant points, PR 24 (1991) 441-451
9. Ray, B.K., Ray, K.S.: An algorithm for detecting dominant points and polygonal approximation of digitized curves, PRL 13 (1992) 849-856
10. Ray, B.K., Ray, K.S.: Detection of significant points and polygonal approximation of digitized curves, PRL 12 (1992) 443-452
11. Cornic, P.: Another look at the dominant point detection of digitized curves, PRL 18 (1997) 13-25
12. Marji, M., Siy, P.: A new algorithm for dominant points detection and polygonization of digital curves, PR 36 (2003) 2239-2251
13. Chung, P.C., Tsai, C.T., Chen, E.L., Sun, Y.N.: Polygonal approximation using a competitive Hopfield neural network, PR 27 (1994) 1505-1512
14. Perez, J.C., Vidal, E.: Optimum polygonal approximation of digitized curves, PRL 15 (1994) 743-750
15. Horng, J.H., Li, J.T., An automatic and efficient dynamic programming algorithm for polygonal approximation of digital curves, PRL 23 (2002) 171-182
16. Yin, P.Y.: A new method for polygonal approximation of digital curves, PRL 19 (1998) 1017-1026
17. Huang, S.C., Sun Y.N.: Polygonal approximation using genetic algorithm, PR 32 (1999) 1409-1420
18. Sarkar, B., Singh, L.K., Sarkar, D.: Approximation of digital curves with line segments and circular arcs using genetic algorithms, PRL 24 (2003) 2585-2595
19. Yin, P.Y.: A new circle/ellipse detector using genetic algorithm, PRL 20 (1999) 731-740
20. Deb, K.: Multi-Objective optimization using Evolutionary algorithms, Wiley, London, 2001
21. Schaffer, J.D., Grefenstette, J.J.: Multiobjective learning via genetic algorithms, In Proceedings of the 9th IJCAI (1985) 593-595
22. Fonseca, C.M., Fleming, P.J.: Genetic algorithm for multi-objective optimization: formulation, discussion and generalization, In the proceedings of the fifth ICGA (1993) 416-423
23. Srinivas, N., Deb, K.: Multiobjective optimization using nondominated sorting in genetic algorithm, EC 2 (1994) 221-248
24. Deb, K., Agrawal, S., Pratab, A., Meyarivan, T., A fast and elitist multi-objective genetic algorithm: NSGA-II, IEEE Transactions on EC 6 (2000) 182-197
25. Knowles, J.D., Corne, D.W.: Approximating the nondominated front using the Pareto archived evolution strategy, EC 8 (2000) 149-172
26. Zitzler, E., Thiele, L.: Multiobjective evolutionary algorithms : a comparative study and the strength pareto approach, IEEE Transactions on EC 3 (1999) 257-271
27. Coello Coello C. A.: A short tutorial on Evolutionary Multiobjective Optimisation, In First International Conference on Evolutionary Multi-Criterion Optimization, Lecture Notes in Computer Science, . Springer-Verlag n° 1993 (2001) 21-40
28. Chafekar, D., Xuan, J., Rasheed, K.: Constrained Multi-objective Optimization Using Steady State Genetic Algorithms, In Proceedings of GECC (2003), 813-824