



A distance measure for molecular structures and its computing method

Eiichi Tanaka^{*}, Kazunori Takemasa, Sumio Masuda

Department of Electrical and Electronics Engineering, Faculty of Engineering, Kobe University, Nada, Kobe 657, Japan

Received 14 April 1997; revised 15 October 1997

Abstract

This paper describes a distance between general line drawings for molecular structures that is essentially the same as the distance function reported by Cox and shows its computing method with a pruning technique to obtain an approximate value. The distance can be applied to comparing molecular structures. © 1998 Elsevier Science B.V. All rights reserved.

Keywords: Pattern matching; Distance; Polygon; Molecular structure

1. Introduction

A similarity measure for 2D or 3D objects and its computing method is one of the basic research topics in pattern recognition with many applications. Some applications in chemistry are found in the literature (e.g., Doucet and Weber, 1996, p. 328; Johnson and Maggiora, 1990, p. 173).

Recently, Cox et al. (1989) defined a distance, that is, a dissimilarity measure, between convex polygons and proved that the distance function is unimodal with respect to parallel transformation for a fixed rotation angle. Therefore, the distance between convex polygons for a fixed rotation angle can be computed by the method of steepest descent. Bloch et al. (1993) extended this measure and the

computing method to the case of convex polyhedra. If we apply this distance function to general line drawings, the distance function becomes multimodal. We should note that the exact similarity value is not always required; an approximate value is sufficient in many applications.

This paper presents a distance function between two general line drawings for chemical structures that is essentially the same as the distance function by Cox et al. (1989) and shows its computing method to obtain an approximate value. A preliminary result was reported in Tanaka et al. (1993).

2. A distance measure between two line drawings

Consider two convex polygons A and B that have m and n vertices, respectively. A and B can be expressed as follows:

$$A = (V_a, E_a), \quad B = (V_b, E_b), \quad (1)$$

^{*} Corresponding author. Present address: 15-16 Ohmori-cho, Nishinomiya-shi, Hyogo-ken 663, Japan.

where V and E are the sets of vertices and edges, respectively. That is, $V_a = \{a_1, a_2, \dots, a_m\}$, $V_b = \{b_1, b_2, \dots, b_n\}$, $E_a = \{(a_i, a_j) | \text{if there is a line between } a_i \text{ and } a_j\}$, and $E_b = \{(b_i, b_j) | \text{if there is a line between } b_i \text{ and } b_j\}$.

Let e be an edge. The distance from a_i to e , denoted by $d(a_i, e)$, is the shortest Euclidean distance among $d(a_i, v)$ s, where v is an arbitrary point on e . Define the distance from a_i to B as follows:

$$\bar{d}(a_i, B) = \begin{cases} \min_{e \in E_b} \{d(a_i, e)\}, & \text{if } a_i \notin \text{Int}(B), \\ 0, & \text{otherwise,} \end{cases} \quad (2)$$

where $\text{Int}(B)$ means the internal region of B .

In this paper we explain a distance measure and its computing method for 2D line drawings. However, it is quite easy to extend them for 3D ones.

Without loss of generality, the origin of the axes is supposed to coincide with the center of gravity of A . Let (x, y) be the coordinates of the center of gravity of B and θ be the rotation angle around (x, y) . Then, B must be written as $B(x, y, \theta)$. Hereafter the notation $A(\mathbf{0})$ is used instead of $A(0, 0, 0)$, if no confusion occurs. Define a distance between $A(\mathbf{0})$ and $B(x, y, \theta)$ as follows:

$$\begin{aligned} \bar{D}(A(\mathbf{0}), B(x, y, \theta)) &= \sum_{i=1}^m \bar{d}(a_i, B(x, y, \theta))^2, \\ \bar{D}(B(x, y, \theta), A(\mathbf{0})) &= \sum_{i=1}^n \bar{d}(b_i, A(\mathbf{0}))^2, \\ \bar{D}(A(\mathbf{0}) : B(x, y, \theta)) &= \bar{D}(A(\mathbf{0}), B(x, y, \theta)) + \bar{D}(B(x, y, \theta), A(\mathbf{0})). \end{aligned} \quad (3)$$

Cox et al. (1989) and Bloch et al. (1993) showed the following.

The distance function $\bar{D}(A(\mathbf{0}) : B(x, y, \theta))$ is convex with respect to x and y for fixed θ , if A and B are convex polygons and polyhedra.

However, $\bar{D}(A(\mathbf{0}) : B(x, y, \theta))$ is not convex with respect to θ for fixed x and y .

A general line drawing in this paper is not only a connected one but also a disconnected one, and is

not only a 2D one but also a 3D one. Hereafter, let A and B be line drawings. In this paper the distance from a_i to $B(x, y, \theta)$ and that from b_j to $A(\mathbf{0})$ are defined as follows:

$$\begin{aligned} d(a_i, B(x, y, \theta)) &= \min_{e \in E_b} \{d(a_i, e)\}, \\ d(b_j, A(\mathbf{0})) &= \min_{e \in E_a} \{d(b_j, e)\}. \end{aligned} \quad (4)$$

We will define the distance from $A(\mathbf{0})$ to $B(x, y, \theta)$ and that from $B(x, y, \theta)$ to $A(\mathbf{0})$ as follows:

$$\begin{aligned} D(A(\mathbf{0}), B(x, y, \theta)) &= \sum_{i=1}^m d(a_i, B(x, y, \theta))^2, \\ D(B(x, y, \theta), A(\mathbf{0})) &= \sum_{i=1}^n d(b_i, A(\mathbf{0}))^2. \end{aligned} \quad (5)$$

Since $D(A(\mathbf{0}), B(x, y, \theta))$ is not always equal to $D(B(x, y, \theta), A(\mathbf{0}))$, we will define the distance between $A(\mathbf{0})$ and $B(x, y, \theta)$ as follows:

$$\begin{aligned} D(A(\mathbf{0}) : B(x, y, \theta)) &= D(A(\mathbf{0}), B(x, y, \theta)) + D(B(x, y, \theta), A(\mathbf{0})). \end{aligned} \quad (6)$$

Since we are considering general line drawings, the distance function $D(A(\mathbf{0}) : B(x, y, \theta))$ is not convex for both “ (x, y) for fixed θ ” and “ θ for fixed (x, y) ”. A possible method of computing $D(A(\mathbf{0}) : B(x, y, \theta))$ is a brute force method with effective priming.

The distance between A and B is defined as follows:

$$D(A : B) = \min_{x, y, \theta} D(A(\mathbf{0}) : B(x, y, \theta)). \quad (7)$$

Hereafter we will mention a characteristic of the distance. Consider the two line drawings in Fig. 1.

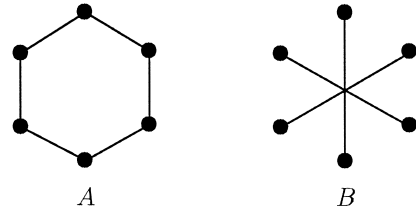


Fig. 1. Two line drawings.

Though the two line drawings are much different, $D(A:B) = 0$. The reason comes from the definition of the distance. That is, the distance is based on the Euclidean distance *from a vertex to an edge*, not from an edge to an edge nor from an edge to a vertex.

In chemical graphs a vertex and an edge denote an atom and a bonding, respectively. An atom is primarily important to compare the shapes of compounds. So our distance seems to be useful at least for comparison of chemical compounds.

3. An approximate computing method

Let $D(A(\mathbf{0}):B(\hat{\theta}))$ be the distance between A and B for fixed θ . That is,

$$D(A(\mathbf{0}):B(\hat{\theta})) = \min_{x,y} D(A(\mathbf{0}):B(x,y,\theta)). \quad (8)$$

Since $D(A(\mathbf{0}):B(x,y,\theta))$ is not convex even for fixed θ , we will compute an approximate value of $D(A(\mathbf{0}):B(\hat{\theta}))$. Let $D(A(\mathbf{0}):B(\hat{\theta})|(x,y))$ be the value computed by the method of steepest, descent starting at (x,y) . Obviously, we have

$$D(A(\mathbf{0}):B(\hat{\theta})) \leq D(A(\mathbf{0}):B(\hat{\theta})|(x,y)). \quad (9)$$

To obtain a more exact value than $D(A(\mathbf{0}):B(j)|(x,y))$, the computation must start at plural points. Define $D(A(\mathbf{0}):B(\hat{\theta})|(x_1,y_1),\dots,(x_h,\dots,y_h))$ as follows:

$$\begin{aligned} & D(A(\mathbf{0}):B(\hat{\theta})|(x_1,y_1),\dots,(x_h,y_h)) \\ &= \min_{i=1,\dots,h} D(A(\mathbf{0}):B(\hat{\theta})|(x_i,y_i)). \end{aligned} \quad (10)$$

Let $\xi = \{(x_1,y_1),\dots,(x_h,y_h)\}$. Notation $D(A(\mathbf{0}):B(\hat{\theta})|\xi)$ is used instead of $D(A(\mathbf{0}):B(\hat{\theta})|(x_1,y_1),\dots,(x_h,y_h))$. Evidently, we have

$$D(A(\mathbf{0}):B(\hat{\theta})) \leq D(A(\mathbf{0}):B(\hat{\theta})|\xi). \quad (11)$$

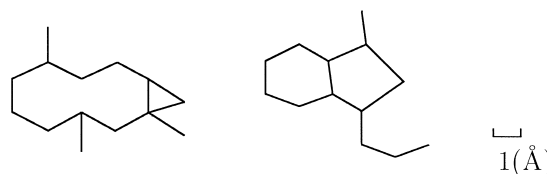


Fig. 2. Two chemical graphs.

The contour map of the distance between the two line drawings of Fig. 2 is shown in Fig. 3. We can observe the behavior of the method of steepest descent in Fig. 3. The map shows that there are at least four minimal values. In all the cases that we computed for chemical graphs, we observed that the minimum value $D(A(\mathbf{0}):B(x,y,\theta))$ is in the region $[-r \leq x \leq r, -r \leq y \leq r]$, where r is the average distance between neighboring atoms. Then, we can use $D(A(\mathbf{0}):B(\hat{\theta})|\xi)$ instead of $D(A(\mathbf{0}):B(\hat{\theta}))$, if we choose ξ appropriately. Note that the larger ξ is, the more exactly the distance D can be obtained but the more computing time is required. That is, there is a trade-off between the exactness of D and the computing time. The point of compromise depends on the problem. In other words there is no good way to choose a suitable ξ a priori. ξ must be determined experimentally. As an example, the starting set ξ that is made of the nine white circles regularly placed in Fig. 3 were satisfactory for odor molecules.

The SDM procedure shown in Scheme 1 computes $D(A(\mathbf{0}):B(\hat{\theta})|\xi)$. $D(A:B|\xi,\Delta\theta)$ in Scheme 1 should be defined as follows:

$$D(A:B|\xi,\Delta\theta) \stackrel{\Delta}{=} \min_{\theta=0,\Delta\theta,\dots,360,-\Delta\theta} D(A(\mathbf{0}):B(\hat{\theta})|\xi). \quad (12)$$

The computing method of $D(A:B|\xi,\Delta\theta)$ based on Eq. (12) is called Method 1.

4. A pruning technique

Let A be a line drawing and A^C be the convex hull that covers A . In Fig. 4 an example of a convex hull of a line drawing is depicted. Consider two line drawings A and B . Let $R(A,B)$ be the common region of A^C and B^C . Let $R(A) = A^C - R(A,B)$ and

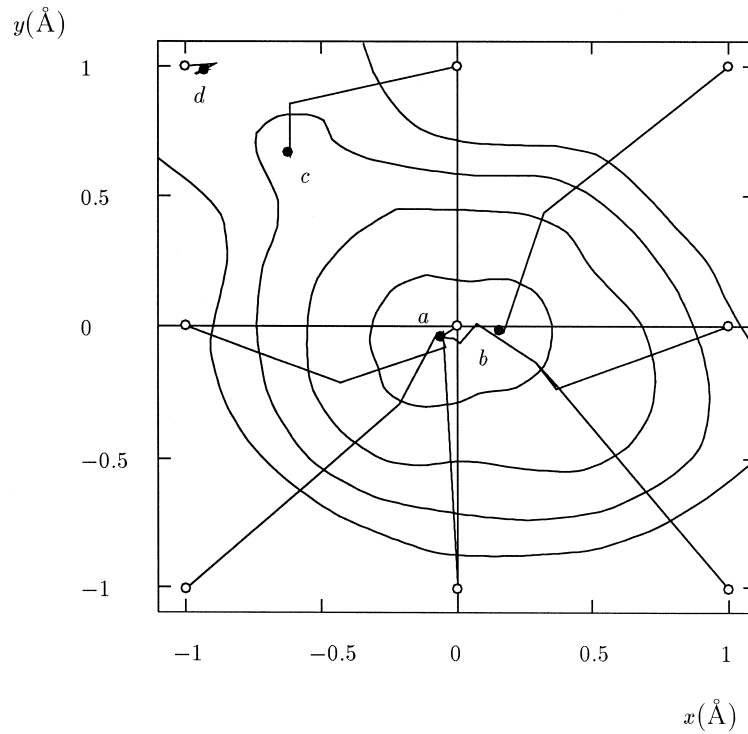


Fig. 3. A contour map of distance between the two line drawings shown in Fig. 2. \circ and \bullet indicate a starting point and a minimal point, respectively. The minimal values at a , b , c and d are 14.1, 14.3, 19.7 and 20.6, respectively.

Input : $A, B, \theta, \xi = \{(x_1, y_1), \dots, (x_h, y_h)\}$

Output : $D(A(0) : B(\hat{\theta}) \mid \xi)$

Procedure $SDM(\theta, \xi, D(A(0) : B(\hat{\theta}) \mid \xi))$

begin

$D_{min} := \infty$;

for $i := 1$ **to** h **do**

begin

compute $D(A(0) : B(\hat{\theta}) \mid (x_i, y_i))$ using the method of steepest descent ;

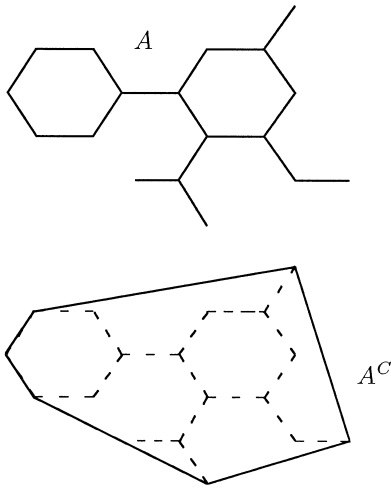
$D_{min} := \min\{D_{min}, D(A(0) : B(\hat{\theta}) \mid (x_i, y_i))\}$;

end

$D(A(0) : B(\hat{\theta}) \mid \xi) := D_{min}$;

end.

Scheme 1. Method 1.

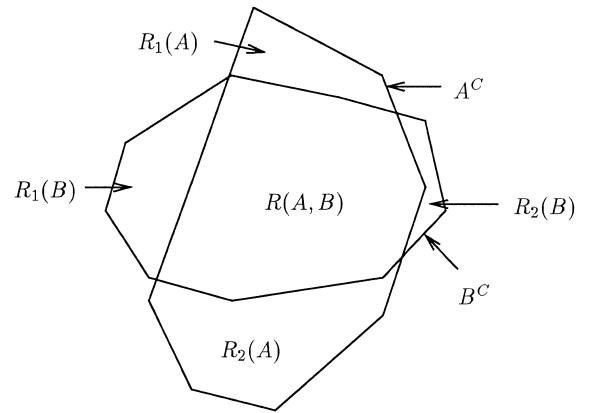
Fig. 4. A line drawing A and its convex hull A^C .

$R(B) = B^C - R(A, B)$. In the example of Fig. 5, $R(A) = R_1(A) \cup R_2(A)$ and $R(B) = R_1(B) \cup R_2(B)$.

$$\begin{aligned}
 D(A(\mathbf{0}) : B(x, y, \theta)) &= \sum_{a_i \in A^C} d(a_i, B(x, y, \theta))^2 + \sum_{b_i \in B^C} d(b_i, A(\mathbf{0}))^2 \\
 &= \sum_{a_i \in R(A)} d(a_i, B(x, y, \theta))^2 \\
 &\quad + \sum_{a_i \in R(A, B)} d(a_i, B(x, y, \theta))^2 \\
 &\quad + \sum_{b_i \in R(B)} d(b_i, A(\mathbf{0}))^2 \\
 &\quad + \sum_{b_i \in R(A, B)} d(b_i, A(\mathbf{0}))^2 \\
 &\geq \sum_{a_i \in R(A)} d(a_i, B(x, y, \theta))^2 \\
 &\quad + \sum_{b_i \in R(B)} d(b_i, A(\mathbf{0}))^2. \tag{13}
 \end{aligned}$$

The following relations are observed:

$$\begin{aligned}
 d(a_i, B(x, y, \theta)) &\geq \bar{d}(a_i, B^C(x, y, \theta)), \\
 \text{if } a_i &\in R(A(\mathbf{0})), \\
 d(b_i, A(\mathbf{0})) &\geq \bar{d}(b_i, A^C(\mathbf{0})), \\
 \text{if } b_i &\in R(B(x, y, \theta)). \tag{14}
 \end{aligned}$$

Fig. 5. Superposing of two convex hulls A^C and B^C .

Then the following inequality holds:

$$D(A(\mathbf{0}) : B(\theta)) \geq \bar{D}(A^C(\mathbf{0}) : B^C(\theta)). \tag{15}$$

Note that $D(A^C(\mathbf{0}) : B^C(x, y, \theta))$ is convex with respect to x, y for fixed θ , since the distance function $\bar{d}(a_i, B^C(x, y, \theta))$ is convex under the same condition. This means that $\bar{D}(A^C(\mathbf{0}) : B^C(\theta))$ for fixed θ can be computed by the method of steepest descent. Let $D(\theta)_{\min}$ be the minimum of $D(A(\mathbf{0}) : B(\phi) | \xi, \Delta\phi)$ for $0 \leq \phi \leq \theta$. If $D(\theta)_{\min} \leq \bar{D}(A^C(\mathbf{0}) : B^C(\theta + \Delta\theta))$ in the process of computing $D(A : B | \xi, \Delta\theta)$, we need not compute $D(A(\mathbf{0}) : B(\theta + \Delta\theta | \xi))$.

The procedure Dist_2 in Scheme 2 is a computing method with pruning for $D(A : B | \xi, \Delta\theta)$. This method is called Method 2. Let $k \cdot \Delta\theta = 360$.

5. The method of valley trekking

Recall that $\bar{D}(A^C : B^C(x, y, \theta))$ has only one minimum value for fixed θ . The increasing and the decreasing of the value of $\bar{D}(A^C : B^C(x, y, \theta))$ are repeated alternately. Then, the following computing method can be considered.

1. In the case that $\bar{D}(A^C : B^C(x, y, \theta))$ is decreasing for increasing θ , we go down the steepest slope. That is, the method of steepest descent can be applied for x, y and θ .
2. In the case that $\bar{D}(A^C : B^C(x, y, \theta))$ is increasing for increasing θ , we go up the gentlest slope. We call this computation the method of gentlest climb.

Input : $A, B, \xi = \{(x_1, y_1), \dots, (x_h, y_h)\}, \Delta\theta$

Output : $D(A : B \mid \xi, \Delta\theta)$

Procedure $Dist_2(\xi, \Delta\theta, D(A : B \mid \xi, \Delta\theta))$

begin

$\theta := 0$;

for $i := 1$ **to** k **do**

begin

$\theta := \theta + \Delta\theta$;

compute $\overline{D}(A^C(\mathbf{0}) : B^C(\hat{\theta}))$ using the method of steepest descent ;

$\overline{D}(A^C : B^C)[i] := \overline{D}(A^C(\mathbf{0}) : B^C(\hat{\theta}))$;

end

$SDM(0, \xi, D(A(\mathbf{0}) : B(\hat{0}) \mid \xi))$;

$D(0)_{min} := D(A(\mathbf{0}) : B(\hat{0}) \mid \xi)$;

$\theta := 0$;

for $i := 1$ **to** k **do**

begin

$\theta := \theta + \Delta\theta$;

if $\overline{D}(A^C : B^C)[i] < D(\theta - \Delta\theta)_{min}$ **then**

begin

$SDM(\theta, \xi, D(A(\mathbf{0}) : B(\hat{\theta}) \mid \xi))$;

$D(\theta)_{min} := \min\{D(\theta - \Delta\theta)_{min}, D(A(\mathbf{0}) : B(\hat{\theta}) \mid \xi)\}$;

end

end

$D(A : B \mid \xi) := D(360)_{min}$;

end.

Scheme 2. Method 2.

Input : $A, B, \alpha, \beta, \Delta\theta$

Output : $\overline{D}(A^C : B^C)$

Procedure *Valley-trekking*($A, B, \overline{D}(A^C : B^C)$)

begin

$x := 0; y := 0; \theta := 0;$

compute $\overline{D}(A^C(0) : B^C(\hat{0}))$ using the method of steepest descent ;

while $\theta < 360$ **do**

begin

compute $(g_x, g_y, g_\theta) = \text{grad}(\overline{D}(A^C(0) : B^C(x, y, \theta)))$;

$x' := x - \alpha g_x$; $y' := y - \alpha g_y$;

if $g_\theta \neq 0$ **then**

$\theta' := \theta + \beta |g_\theta|$

else $\theta' = \theta + \Delta\theta$;

compute $\overline{D}(A^C(0) : B^C(x', y', \theta'))$;

$D_{\min} := \min\{D_{\min}, \overline{D}(A^C(0) : B^C(x', y', \theta'))\}$;

end ;

$\overline{D}(A^C : B^C) := D_{\min}$;

end.

Scheme 3. Method 3.

Since the two methods (1) and (2) are similar to valley trekking, we call the combination of the above two methods the method of valley trekking. Let α and β be step sizes.

Scheme 3 shows the computing method. Method 3 is the method by which valley-trekking is used in the process of pruning of Method 2.

6. Experimental results

We carried out a computer experiment to compare 62 chemical graphs (i.e., 1891 pairs). The average

Table 1
Average computing time (s)

Method	Step angle $\Delta\theta$		
	1°	2°	3°
1 (t_1)	354.6	177.8	118.0
2 (t_2)	101.7	50.9	34.3
3 (t_3)	88.6	44.8	30.3
3 (t_3/t_1)	0.248	0.252	0.257
3 (t_3/t_2)	0.871	0.881	0.848

Method 1, Method 2 and Method 3 are the methods stated in Section 3, Section 4 and Section 5, respectively. t_x is the computing time of Method x .

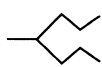
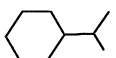
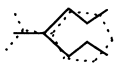
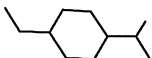
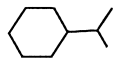
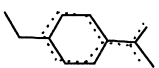
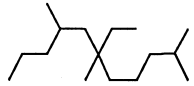
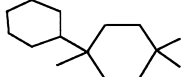
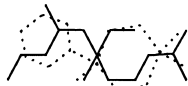
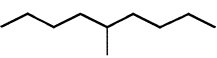
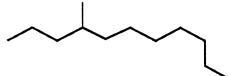
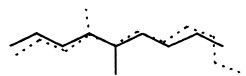
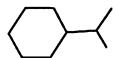
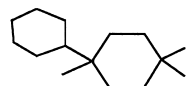
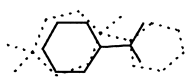
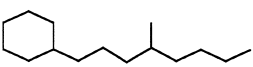
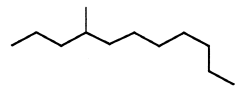
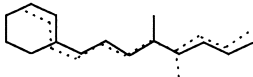
line drawings	matching	distance
 		5.18
 		7.18
 		9.42
 		9.79
 		9.89
 		10.70

Fig. 6. Examples of matchings.

computing times of the three methods are shown in Table 1. The computer used was a SUN SPARC station 4 and the program was written in Pascal. The matchings and their distances of six pairs of chemical graphs are shown in Fig. 6.

7. Concluding remarks

The distance measure between chemical structures proposed in this paper is essentially the same as the distance measure between convex polygons (Cox et al., 1989). This distance function is multimodal. Therefore, the standard method we can use is a brute force method. However, as we have shown, the method of valley-trekking can be applied effectively in pruning. This distance measure can be applied to comparing chemical structures. A similarity or distance measure between general line drawings for pattern recognition (e.g., Matsuyama et al., 1983) has not been fully discussed and still remains a future problem.

Acknowledgements

We would like to thank Mr. Hiroaki Awano for making a computer program at the early stage of this

research, Dr. Isabel Bloch and Dr. Henri Maître for sending us their paper and anonymous reviewers for precious comments.

References

- Bloch, I., Maître, H., Minotix, M., 1993. Optimal matching of 3-D convex polyhedra with applications to pattern recognition. *Pattern Recognition and Image Analysis* 3, 137–149.
- Cox, P., Maitre, H., Minoux, M., Ribeiro, C., 1989. Optimal matching of convex polygons. *Pattern Recognition Letters* 9, 327–334.
- Doucet, J.P., Weber, J., 1996. Molecular similarity. In: *Computer-Aided Molecular Design*. Academic Press, New York.
- Johnson, M.A., Maggiora, G.M., 1990. *Concepts and Applications of Molecular Similarity*. Wiley, New York.
- Matsuyama, T., Arita, E., Nagao, M., 1983. A structural matching of line drawings using spatial relations between line segments. *J. Information Processing Society of Japan* 24, 735–744.
- Tanaka, E., Awano, H., Masuda, S., 1993. A proximity measure of line drawings for comparison of chemical compounds. In: *Proceedings of the 5th International Conference, CAIP'93*. pp. 291–298.