JOURNAL OF ALGORITHMS 7, 277–284 (1986)

Bipartite Graph Matching for Points on a Line or a Circle

MICHAEL WERMAN,* SHMUEL PELEG,* ROBERT MELTER,† AND T. Y. KONG‡§

Center for Automation Research, University of Maryland, College Park, Maryland 20742

Received September 15, 1984

Given two sets of M points on a line or on a circle, a minimal matching between them is found in $O(M \log M)$ time. The circular case, where the distance between two points is the length of the shortest arc connecting them, is shown to have the same complexity as the simpler linear case. Finding the shift of one of the sets, linear or circular, that minimizes the cost of matching is also discussed. © 1986 Academic Press, Inc.

1. INTRODUCTION

Let $A = \{a_1, a_2, \ldots, a_M\}$, $B = \{b_1, b_2, \ldots, b_M\}$ be two sorted multisets of points on a line. A matching of A and B is a bijection of A onto B, and is uniquely determined by a permutation π of $\{1, 2, \ldots, M\}$ such that a_i is paired with $b_{\pi i}$. In the sequel the matching itself will sometimes be designated π. Define the cost of a matching to be the sum of the distances of all pairs in it. It can be seen that when $\{a_i\}$ and $\{b_i\}$ are sorted, a minimal cost matching is determined by the identity permutation, where the pairing is $\langle a_1, b_1\rangle, \ldots, \langle a_M, b_M\rangle$.

*Permanent address: Dept. of Computer Science, The Hebrew University of Jerusalem, 91904 Jerusalem, Israel. These authors were supported in Israel by a grant from the Bergman Foundation.

†Permanent address: Dept of Mathematics, Long Island University, Southampton, NY 11968.

‡Permanent address: Programming Research Group, Oxford University Computing Laboratory, England.

§The support of the National Science Foundation under Grant DCR-82-18408 is gratefully acknowledged, as is the help of Janet Salzman in preparing this paper.

0196-6774/86 $3.00

The matching problem is more interesting when A and B are sets of points on the unit circle, and the distance between two points is taken to be the length of the shortest arc between them (or equivalently, the angle between them). In [1] a minimal matching for this case is approximated by examining only all circular permutations; the time complexity turns out to be $O(M^2)$. It was later shown [2] that this "approximate" minimal matching is in fact minimal. In the present paper, however, an $O(M \log M)$ algorithm is presented for finding the minimal matching, and this is shown to be a lower bound for the two matching problems.

Computing the minimal matching is important in pattern recognition, when feature sets of different objects are compared. Features can be gray level histograms in digital pictures (a linear feature), or histograms of gradient directions in textures (a circular feature). Another problem in pattern recognition is point pattern matching, where given two point sets a translation is sought to minimize the distance between them. We can define the distance between two point sets to be the cost of their minimal matching. Using this definition, the translation which minimizes the distance between two linear point sets, and the rotation which minimizes the distance between two circular point sets, are discussed in Section 3.

2. Optimal Matching

Given two point sets, the minimum cost matching between the sets is sought. In this paper we discuss the cases where the points are on a line, i.e., real numbers, or on a circle, i.e., angles between 0 and 2π.

LEMMA 1. *If the distance d between points in A and B satisfies the triangle inequality then the cost of a minimal matching of A and B equals the cost of a minimal matching of A' and B', where A' and B' are respectively the largest submultisets of A and B such that A' and B' are disjoint.*

Let $a_i \in A$, $b_j \in B$, and $a_i = b_j$. Suppose that there is a minimal cost matching in which $\langle a_i, b_k \rangle$ and $\langle a_l, b_j \rangle$ are members for some k and l. From the triangle inequality

$$d(a_l, b_j) + d(a_i, b_k) \geq d(a_l, b_k) + 0 = d(a_l, b_k) + d(a_i, b_j).$$

Therefore, pairing up equal elements will never add to the cost of a matching.

Since linear and circular distance are metrics and satisfy the triangle inequality, whenever we wish to find the cost of a minimal matching, pairing all equal elements reduces the size of the problem without sacrificing minimality. In the sequel we will assume without loss of generality that the two multisets are disjoint.

LEMMA 2. *$\Omega(M \log M)$ is a lower bound for the time it takes to find the minimal cost matching in both the linear and the circular cases.*

The set equality problem which has an $\Omega(M \log M)$ lower bound in many computational models [3] can be reduced to minimal matching, since two sets of numbers are equal iff the cost of their minimal matching is 0. This shows that computing the minimal matching takes at least $\Omega(M \log M)$.

2.1. *Linear Matching*

It has been shown [1, 2] that for two given sorted sets A, B on a line, a matching π which minimizes the distance

$$d = \sum_{i=1}^{M} |a_i - b_{\pi i}|$$

is the identity, and the pairs are $\langle a_1, b_1 \rangle, \langle a_2, b_2 \rangle, \dots, \langle a_M, b_M \rangle$. This can be seen by noticing that the claim is true for two sets of two elements each. For larger sets, examine any two pairs $\langle a_i, b_{\pi i} \rangle$ and $\langle a_{i+1}, b_{\pi(i+1)} \rangle$ in a minimal matching. Their distance $|a_i - b_{\pi i}| + |a_{i+1} - b_{\pi(i+1)}|$ must be minimal as part of the minimal matching. If $b_{\pi i} > b_{\pi(i+1)}$ we can exchange them without adding to the cost of the matching, yielding a new permutation π. Continuing in this way π will eventually become the identity. If the sets are initially sorted the complexity of computing the distance is $O(M)$, while it is $O(M \log M)$ otherwise.

2.2. *The Circular Case*

Let $A = \{a_1, a_2, \dots, a_M\}$, $B = \{b_1, b_2, \dots, b_M\}$ be two sorted sets of points on the unit circle, $0 \leq a_i, b_i < 2\pi$. Let $C = \{c_1, c_2, \dots, c_{2M}\}$ be the sorted union of A and B. To simplify the determination of the minimal matching, we define the following functions:

$N_a(\alpha)$: the number of points a_i such that $a_i < \alpha$,

$N_b(\alpha)$: the number of points b_i such that $b_i < \alpha$,

and

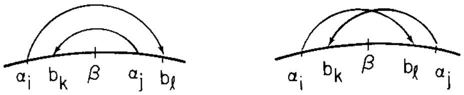
$$F(\alpha) = N_a(\alpha) - N_b(\alpha). \tag{1}$$

$F(\alpha)$ has discontinuities only at points $c_i \in C$, and is piecewise constant. The direction of an arc connecting a_i to $b_{\pi i}$ is taken to be from a_i to $b_{\pi i}$; clockwise arcs are considered positive and counterclockwise arcs negative.

DEFINITION. Call a matching *unidirectional if* all arcs passing a point have the same orientation.

LEMMA 3. *A minimal matching is unidirectional.*

Graphically, the only possibilities for two arcs passing the point β in different directions are as follows:



None of these cases can be in a minimal matching, since the matching $\langle a_i, b_k \rangle, \langle a_j, b_l \rangle$ will have a lower cost in both cases with no overlap between arcs in opposite directions. □

Let K be the number of arcs passing the point 0 in a minimal matching (K will be computed later). For positive K all these arcs are clockwise, for negative K they are all counterclockwise, and when $K = 0$ no arcs pass the point 0. Lemma 3 assures that all arcs passing zero are in the same direction. The function $F(\alpha) + K$ gives the number of arcs passing the point α in a minimal matching. This is true because when we encounter a point $a_i \in A$ in a clockwise traversal of the circle, either a counterclockwise arc ("negative" arc) is ending, or a clockwise ("positive") arc is beginning. In both cases, the count is increased by 1 as is the case with $F(\alpha)$. When several points a_i are equal, the function F increases by their number. When a point $b_i \in B$ is encountered the count is decreased by 1. When several points b_i are equal, the function F decreases by their number.

LEMMA 4. *The cost of a minimal matching given that K arcs cross point* 0 *is*

$$C(K) = \int_0^{2\pi} |F(x) + K| \, dx.$$

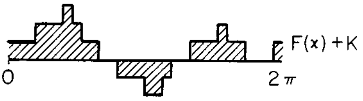
This is correct since $|F(x) + K|$ is the number of arcs passing x, and thus the integral gives the sum of lengths of all arcs. A K minimizing the cost of the matching is sought.

LEMMA 5. *The function* $C(K) = \int_0^{2\pi} |F(x) + K| \, dx$ *is minimized when* K *satisfies*

$$\int_{F(x) > -K} dx = \int_{F(x) < -K} dx;$$

$C(K)$ is the area bounded by $F(x) + K$. When K is increased by a small value $\varepsilon > 0$, additions to this area occur, where $F(x) + K > 0$ (addition of

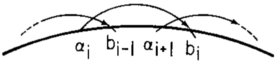
$\varepsilon \cdot \int_{F(x)+K>0} dx$), and the area gets smaller by $\varepsilon \cdot \int_{F(x)+K<0} dx$, where $F(x) + K < 0$. Since $\int_{F(x)+K>0} dx$ is monotonic non-increasing in K, and $\int_{F(x)+K<0} dx$ is monotonic non-decreasing, the minimum can only be reached when the two values are equal.



Once K is known, starting from a point where $F(\alpha) + K = 0$ we can pair the points in order, as in the linear case. This is equivalent to "opening" the circle into a line by cutting it at a point where no arc passes, and then using the linear matching. Before proceeding, we need to prove the existence of a point not included in any arc.

LEMMA 6. *In a minimal matching there is at least one point which is not a point of any arc, i.e., the function $F(x) + K$ is zero in some interval (c_i, c_{i+1}).*

Suppose there is a minimal matching in which every point is interior to some arc. From Lemma 3 it follows that if every point is interior to some arc, all arcs will have the same direction. Within every arc $\langle a_i, b_i \rangle$ we must have at least one entering arc and one exiting arc in order to cover the entire circle.



We can find a subset of that minimal matching such that within each arc there is exactly one entry and one exit. Denote this subset by $\langle a_1, b_1 \rangle, \ldots, \langle a_T, b_T \rangle$; let the one arc entering $\langle a_i, b_i \rangle$ terminate at b_{i-1}, and let the one exiting arc start with a_{i+1} (see the diagram). In this case if we replace that subset by $\langle a_{(i+1) \bmod T}, b_i \rangle$ for all i, a lower cost matching is obtained.

Finding a minimal matching can be summarized as follows:

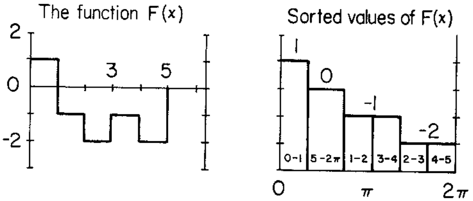
1. Compute $F(x)$ of Eq. (1) at all points c_i. The value of $F(x)$ in the segment $(c_i, c_{i+1}]$ is equal to $F(c_i)$.

2. Sort the values of $F(c_i)$. Remember the segment associated with each value.

3. Add up the lengths of the segments associated with the sorted values of $F(c_i)$ ($|c_i - c_{i+1}|$), starting with the highest value of F, until the sum exceeds π. Take $-K$ to be the last value of F, $F(c_i)$.

Once K is found, starting with a point $F(x) + K = 0$ or $F(x) = -K$ we can simply match the elements in the sets sequentially moving clockwise.

EXAMPLE. Let $A = \{0, 3, 5, 5\}$ and $B = \{1, 1, 2, 4\}$ be two sets of points between 0 and 2π; then $C = \{0, 1, 1, 2, 3, 4, 5, 5\}$. After sorting the values of F and adding up the respective segments, we find that we pass π when adding a segment for which $F = -1$. Therefore, linear matching can be done if we "open" the circle within the segments (1, 2) or (3, 4). The minimal matching obtained by "opening" in either segment is $\{\langle 3, 2\rangle, \langle 5, 4\rangle, \langle 5, 1\rangle, \langle 0, 1\rangle\}$, with a cost of $1 + 1 + (2\pi - 4) + 1 = 2\pi - 1$.



3. OPTIMAL SHIFTS

3.1. *Linear Shifts*

In this case, given two sets of points $A = \{a_1, a_2, \ldots, a_M\}$ and $B = \{b_1, b_2, \ldots, b_M\}$, a translation t of A is sought that minimizes the distance between the sets. Since the optimal matching will always be $\langle a_1, b_1\rangle, \ldots, \langle a_M, b_M\rangle$, the distance between the two sets under the translation t is

$$C(t) = \sum_{i=1}^{M} |a_i + t - b_i| = \sum_{i=1}^{M} |(a_i - b_i) + t|.$$

It is known that the value of t that minimizes this function is the median of $\{a_i - b_i\}$. It follows from Lemma 7, using a matching argument.

LEMMA 7. *In an optimal translation for the above cost function, the absolute difference between the number of arcs* $\langle a_i, b_i\rangle$ *going right*

$(a_i - b_i < 0)$ and the number of arcs going left $(a_i - b_i > 0)$ is less than or equal to the number of null arcs $(a_i - b_i = 0)$.

A minute shift of A by dx to the right will decrease the cost of a matching by

$$dx^*[\#(\text{right arcs}) - \#(\text{left arcs}) - \#(\text{null arcs})]$$

and similarly to the left. In an optimal matching the decrease cannot be negative, and the number of null arcs has to be greater than or equal the absolute value of the difference between the number of right and left arcs.

The time complexity of finding the optimal translation is thus $O(M)$ when the two sets are initially sorted, and $O(M \log M)$ otherwise.

Other definitions for the cost of a matching are possible. Consider the total cost of a matching to be the sum of all squared pairwise distances:

$$S(t) = \sum_{i=1}^{M} (a_i + t - b_i)^2 = \sum_{i=1}^{M} [(a_i - b_i) + t]^2.$$

By setting the derivatives to zero we can find that the above distance is minimized for $t = -(\Sigma(a_i - b_i))/M$. Equivalently, this distance is minimized when the sets are translated so that both centers of gravity coincide. This translation, however, is independent of the matching, and is optimal for all matchings.

3.2. *Circular Shift*

The algorithm in Section 2.2 constructs a minimal matching of the form $\langle a_i, b_{(i+K)\bmod M}\rangle$. Therefore, for any rotation ϕ of the set A, there is a K such that $\langle (a_i + \phi)\bmod 2\pi, b_{(i+K)\bmod M}\rangle$ is a minimal matching.

Examine all M matchings of the form $\langle a_i, b_{(i+K)\bmod M}\rangle$. For each such matching, "open" the A circle at a_1 and the B circle at b_{K+1} into a line. A is left intact in this opening, and B is being changed into B' such that

$$\begin{aligned} b'_i &\equiv b_{i+k}, & 1 \le i \le M - K, \\ &\equiv b_{i+K-M} + 2\pi, & M - K < i \le M. \end{aligned}$$

Find an optimal shift of the set A using the linear shift algorithm of Section 3.1 on the sets A and B'. An optimal shift will be one giving the lowest cost among all M shifts found. Since finding each linear shift has complexity $O(M)$, finding the optimal rotation has complexity of $O(M^2)$.

ACKNOWLEDGMENTS

The authors would like to thank Azriel Rosenfeld and Marc Snir for their helpful comments.

REFERENCES

1. H. C. CHEN AND A. K. C. WONG, Generalized texture representation and metric, *Comput. Vision Graphics Image Process.* **23**, 1983, 182–206.
2. M. WERMAN, S. PELEG, AND A. ROSENFELD, A distance metric for multidimensional histograms, CAR-TR-90, Center for Automation Research, University of Maryland, College Park, Md., August 1984.
3. M. BEN-OR, Lower bounds for algebraic computation trees, *in* "Proc. 15th ACM Sympos. on Theory of Comput.," pp. 80–86, 1983.