Theoretical Computer Science 66 (1989) 157–180
North-Holland

TESTING THE NECKLACE CONDITION FOR SHORTEST TOURS AND OPTIMAL FACTORS IN THE PLANE

Herbert EDELSBRUNNER*
Department of Computer Science, University of Illinois at Urbana-Champaign, 1304 W. Springfield Avenue, Urbana, IL 61801, U.S.A.

Günter ROTE**
Institut für Mathematik, Technische Universität Graz, Kopernikusgasse 24, A-8010 Graz, Austria

Ermo WELZL
Fachbereich Mathematik, Freie Universität Berlin, Arnimallee 2-6, D-1000 Berlin 33, Fed. Rep. Germany

Abstract. A tour $\mathcal{T}$ of a finite set P of points is a *necklace-tour* if there are disks with the points in P as centers such that two disks intersect if and only if their centers are adjacent in $\mathcal{T}$. It has been observed by Sanders that a necklace-tour is an optimal traveling salesman tour.

In this paper, we present an algorithm that either reports that no necklace-tour exists or outputs a necklace-tour of a given set of n points in $O(n^2 \log n)$ time. If a tour is given, then we can test in $O(n^2)$ time whether or not this tour is a necklace-tour. Both algorithms can be generalized to f-factors of point sets in the plane. The complexity results rely on a combinatorial analysis of certain intersection graphs of disks defined for finite sets of points in the plane.

1. Introduction and definitions

The *traveling salesman problem* is one of the most important problems in the area of combinatorial optimization. It asks for computing a *tour* in a weighted graph (that is, a cycle that visits every vertex exactly once) such that the sum of the weights of the edges in this tour is minimal; such a tour will be called an *optimal tour.*

The traveling salesman problem is known to be NP-complete (see [14]) which currently implies that no algorithm is known which finds an optimal tour in polynomial time. There are generally two approaches to circumvent this difficulty: one is the design of algorithms which compute tours that are close to optimal, the other identifies restricted classes of the problem for which efficient solutions are possible.

In this paper, we are concerned with the *traveling salesman problem* for complete undirected weighted graphs fulfilling the triangle inequality. A particular subclass is the *Euclidean traveling salesman problem* in the plane where the considered graph $\mathcal{G}$ is the *complete distance graph of a* finite set P of points in the plane: the vertices

* Supported by Amoco Found. Fac. Dev. Comput. Sci. 1-6-44862.
** Supported by the Austrian Science Foundation (Fonds zur Förderung der wissenschaftlichen Forschung), Project S32/01.

of $\mathcal{G}$ are the points in P and the weight of an edge between two vertices is the Euclidean distance between the corresponding points. Although the Euclidean traveling salesman problem is a restricted version of the general traveling salesman problem, it has been shown that it is still NP-hard (see [22]). There are a few natural classes of point sets, however, which allow for an efficient construction of optimal tours. For example, the edges on the boundary of the convex hull of P form the unique optimal tour of P if all points of P are extreme.

The class of point sets considered in this paper is based on the notion of the necklace condition.

> Let P be a finite point set in the plane. A tour $\mathcal{T}$ of P is a *necklace-tour* if it is the intersection graph of a set S of closed disks centered at the points in P. If P allows a necklace-tour, then we say that P *satisfies the necklace condition.*

Figure 1 shows an example of a necklace-tour. Reinhold [25] and Sanders [27] showed that the traveling salesman problem becomes easy if a necklace-tour exists (see also [30]). In particular, the following is true.

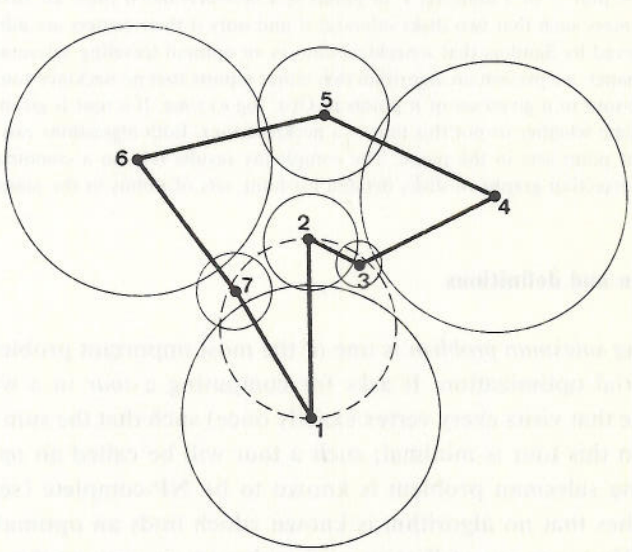

Fig. 1. A necklace-tour.

Proposition 1.1. *A necklace-tour of a finite point set P is the unique optimal tour of P.*

It is easy to construct point sets that do not satisfy the necklace condition (see Fig. 3 and the second remark after Theorem 2.5).

Remark. Note that points 1 and 2 in Fig. 1 are not adjacent in the Delaunay triangulation of the point set shown, since every circle through points 1 and 2 encloses point 3 or point 7 (see [23] for a definition of Delaunay triangulations).

Hence, Fig. 1 and Proposition 1.1 constitute a simple counter-example to the conjecture that every optimal tour of a planar point set is a subgraph of the Delaunay triangulation (see [28] for the first mention of this problem and [15] for the first settlement of the problem; in contrast to the construction shown in Fig. 1, his solution is based on points in rather special position). Notice that the same type of argument can be used to show that optimal m-factors, m any positive integer, are not always subgraphs of the Delaunay triangulation.

Although a necklace-tour of a point set solves the traveling salesman problem for this set, no algorithm has been known that recognizes point sets which satisfy the necklace condition and that computes a necklace tour, if it exists. This paper offers an algorithm that checks a given tour for being a necklace-tour in $O(n^2)$ time and $O(n)$ space, and it presents an algorithm that tests whether a necklace-tour exists and constructs it, if it exists, in $O(n^2 \log n)$ time and $O(n)$ space.

Although our investigations emerged from the study of tours of point sets in the plane, some notions and results are also valid for f-factors and for general graphs for which the triangle inequality holds. In the remainder of this section we present some basic definitions and we give an outline of the paper.

Let $\mathcal{G} = (P, G)$ be a complete weighted graph, with set of vertices $P = \{p_1, p_2, \ldots, p_n\}$ and set of edges G. Throughout this paper, we will use corresponding script and roman letters to denote a graph and its edge set. The length of an edge $\{p_i, p_j\}$ is denoted d_{ij}. All graphs that we deal with fulfill the triangle inequality $d_{ij} + d_{jk} \geqslant d_{ik}$ for all p_i, p_j, p_k in P. In particular, this implies $d_{ij} \geqslant 0$ for all $p_i, p_j \in P$.

A subgraph $\mathcal{F}$ of $\mathcal{G}$ with vertex set P is called *realizable* if for each point p_i there exists a real number r_i, called its *radius*, such that

$$(1) \quad \begin{cases} r_i + r_j \geqslant d_{ij}, & \text{if } \{p_i, p_j\} \in F, \text{ and} \\ r_i + r_j < d_{ij} & \text{if } \{p_i, p_j\} \notin F. \end{cases}$$

The numbers r_i are said to *realize* $\mathcal{F}$.

If $\mathcal{G}$ is the complete distance graph of a set of points P in the Euclidean plane and the r_i are all positive, then these numbers can be interpreted as radii of closed disks centered at the points p_i. In this case, $\mathcal{F}$ is the intersection graph of these disks, that is, $\mathcal{F}$ has an edge between two points if and only if the corresponding disks intersect. If the radii r_i are positive, then we say that $\mathcal{F}$ is *realizable by positive radii.* We will see that some classes of realizable graphs are always realizable by positive radii if they are at all realizable.

If $f = (f_1, f_2, \ldots, f_n)$ is a sequence of integers, a subgraph $\mathcal{F}$ of $\mathcal{G}$ is called an *f-factor* if the degree of p_i in $\mathcal{F}$ is f_i. If $f_1 = f_2 = \cdots = f_n = m$, then we call $\mathcal{F}$ an *m-factor.* In particular, 1-factors are complete matchings, and 2-factors are coverings of vertices by disjoint cycles. 2-Factors are important to us because tours are just connected 2-factors.

The *weight* of a subgraph $\mathcal{F}$ of $\mathcal{G}$ is the sum of the weights of its edges, and an *optimal* f-factor, tour, etc. of $\mathcal{G}$ is an f-factor, tour, etc. with minimum weight. By a 2-factor, tour, etc. of a set of points P in the plane we always mean a 2-factor, tour, etc. of the complete distance graph of P.

Let m be a positive integer smaller than n. For each point $p \in P$, we let $d^{(m)}(p)$ denote the distance from p to the m-nearest neighbor of point p, that is, $d^{(m)}(p)$ is the smallest real number δ such that the cardinality of $\{q \in P \mid d(p, q) \leq \delta\}$ is at least $m+1$. The graph $\mathscr{G}^{(m)}$ is the graph realized by the numbers $d^{(m)}(p)$. In $\mathscr{G}^{(m)}$, every vertex has at least degree m; therefore, the number of edges is at least $\frac{1}{2}mn$. Again, if P is a set of points in the plane, we can define $D^{(m)}(p)$ to be the closed disk centered at p with radius $d^{(m)}(p)$ in which case $\mathscr{G}^{(m)}$ is the intersection graph of these disks.

Below, we give an outline of the structure of this paper and we review the results to be presented. There are algorithms for two kinds of problems considered in this paper.

Problem A. Test the realizability of a given f-factor $\mathscr{F}$ and find radii that realize $\mathscr{F}$, if they exist.

Problem B. Check whether a graph has a realizable f-factor and find it together with the radii that realize it, if they exist.

Section 2 gives a characterization of realizable f-factors based on the relation between the integer programming formulation of the optimal f-factor problem and its linear programming relaxation: a graph has a realizable f-factor if and only if the linear program has an integral and unique optimal solution which is then the unique realizable f-factor. Based on this characterization, the test of existence and the construction of a realizable f-factor (Problem B) can be reduced to the construction of optimal f-factors in bipartite graphs. The latter problem can be performed in a simple and efficient way by network flow techniques. This is described in Section 6.

Section 5.1 describes how the realizability of a given f-factor (Problem A) can be tested by efficiently solving the system of linear equalities (1). As a consequence of the latter results, we derive in Section 5.3 a combinatorial characterization of realizable f-factors which is an extension of the characterization of optimal f-factors by means of alternating cycles.

In Section 2, we also show that a realizable m-factor of a graph is a subgraph of $\mathscr{G}^{(m)}$. This serves as a motivation to show that $\mathscr{G}^{(m)}$ of a set of points in the plane has only $O(mn)$ edges. This is done in Section 3. In Section 5.2, we put these results together and describe how $\mathscr{G}^{(m)}$ can be constructed efficiently. Thus, for m-factors of points in the plane, the running times of the algorithms can be improved over what is needed for non-Euclidean graphs. The running time for Problem A is $O(mn^2)$ and for Problem B it is $O(n^2 \log m(m+\log n))$. The space complexity is $O(mn)$ for both algorithms.

2-Factors play an important role in our algorithms. We will see in Section 2 that realizable 2-factors are unique and optimal. Since tours are 2-factors and necklace-tours are realizable, this shows that the necklace condition for a point set P is satisfied if and only if the optimal 2-factor of P is realizable and is a tour. Thus, for testing the necklace condition, we simply have to find the (unique, optimal) realizable 2-factor, if it exists, and finally check whether it is a tour. This yields our claimed complexity bounds of $O(n^2 \log n)$ time and $O(n)$ space for testing the necklace condition.

Finally in Section 7, we mention possible extensions and further applications of our results, and we present computational experiments pertaining to the usefulness of our methods in practice.

2. Properties of realizable f-factors

In Section 2.1, we state a necessary and sufficient condition for a graph to have a realizable f-factor, and we discuss the consequences of this condition to necklace-tours in Section 2.2. In Section 2.3, we show that a realizable m-factor is a subgraph of $\mathscr{G}^{(m)}$, which is of interest because $\mathscr{G}^{(m)}$ of n points in the plane has only $O(mn)$ edges. The latter result will be shown in Section 3.

2.1. *Characterization of realizable f-factors*

To state our characterization, we need to formulate the optimal f-factor problem as an integer program. Let $\mathscr{G} = (P, G)$ be a weighted graph with $P = \{p_1, p_2, \ldots, p_n\}$. We define a binary variable x_{ij} for each edge $\{p_i, p_j\}$ of the graph, where $x_{ij} = 1$ if $\{p_i, p_j\}$ belongs to the f-factor, and $x_{ij} = 0$ if it does not. The f-factor problem can now be formulated as follows:

$$\text{(F)} \begin{cases} \text{minimize} \displaystyle\sum_{\{p_i,p_j\}\in G} x_{ij} d_{ij} & \text{(F.1)} \\ \text{subject to} \displaystyle\sum_{\{p_i,p_j\}\in G} x_{ij} = f_i \quad \text{for } i = 1, 2, \ldots, n & \text{(F.2)} \\ \text{and } x_{ij} \in \{0, 1\} \qquad \text{for } \{p_i, p_j\} \in G & \text{(F.3)} \end{cases}$$

The linear programming relaxation of this problem, called the *fractional f-factor problem* (FF) is obtained by replacing the constraints (F.3) by

$$\text{(FF.3)} \quad 0 \leqslant x_{ij} \leqslant 1 \quad \text{for } \{p_i, p_j\} \in G,$$

for real numbers x_{ij}.

Below, we present a characterization of when a graph has a realizable f-factor for a given vector f. Throughout this paper, we will use $\mathscr{F}$ to denote an f-factor and F to denote is edge set.

Theorem 2.1. *A graph $\mathcal{G}$ has a realizable f-factor $\mathcal{F}$ if and only if the fractional f-factor problem defined by* (F.1), (F.2), *and* (FF.3) *has a unique optimal solution which is integral. $\mathcal{F}$ is then the f-factor corresponding to this solution.*

Proof. (i) First we prove the "only-if" part of Theorem 2.1. Let $r_1, r_2, \ldots, r_n$ be radii that realize $\mathcal{F}$, that is, they fulfill (1). Then it is clear that these radii can be made to fulfill

$$(1') \quad \begin{cases} r_i + r_j > d_{ij} & \text{if } \{p_i, p_j\} \in F, \text{ and} \\ r_i + r_j < d_{ij} & \text{if } \{p_i, p_j\} \notin F, \end{cases}$$

for example by adding $\frac{1}{3}\delta$ to all radii, where $\delta = \min\{d_{ij} - r_i - r_j \mid \{p_i, p_j\} \notin F\} > 0$. Now assume that $r_1, r_2, \ldots, r_n$ fulfill (1'). If we replace the distances d_{ij} in the objective function (F.1) by $d'_{ij} := d_{ij} - r_i - r_j$, then we get

$$\begin{aligned} \sum_{\{p_i, p_j\} \in G} x_{ij} d'_{ij} &= \sum_{\{p_i, p_j\} \in G} x_{ij}(d_{ij} - r_i - r_j) \\ &= \sum_{\{p_i, p_j\} \in G} x_{ij} d_{ij} - \sum_{\{p_i, p_j\} \in G} x_{ij} r_i - \sum_{\{p_i, p_j\} \in G} x_{ij} r_j \\ &= \sum_{\{p_i, p_j\} \in G} x_{ij} d_{ij} - \sum_{i=1}^{n} \sum_{\{p_i, p_j\} \in G} x_{ij} r_i \\ &= \sum_{\{p_i, p_j\} \in G} x_{ij} d_{ij} - \sum_{i=1}^{n} f_i r_i, \end{aligned}$$

by (F.2). Thus, the new objective function

$$(\text{F.1}') \quad \text{minimize} \sum_{\{p_i, p_j\} \in G} x_{ij} d'_{ij}$$

differs from the old one by an additive constant, and the optimal solutions are the same with respect to both objective functions. But we have

$$d'_{ij} = d_{ij} - r_i - r_j < 0 \quad \text{for } \{p_i, p_j\} \in F \quad \text{and}$$

$$d'_{ij} = d_{ij} - r_i - r_j > 0 \quad \text{for } \{p_i, p_j\} \notin F.$$

Therefore, the unique optimal solution of (F.1') subject to (FF.3) is the solution corresponding to $\mathcal{F}$:

$$x_{ij} = 1 \quad \text{for } d'_{ij} < 0 \ (\{p_i, p_j\} \in F), \quad \text{and}$$

$$x_{ij} = 0 \quad \text{for } d'_{ij} > 0 \ (\{p_i, p_j\} \notin F),$$

and this solution also fulfills (F.2) and is thus the unique optimal solution of (F.1), (F.2), and (FF.3).

(ii) Now, we prove the "if" part of Theorem 2.1. Assume that vector $\hat{x}$ is the unique optimal solution of (F.1), (F.2), and (FF.3). The following rather technical consideration shows that the coefficients d_{ij} in the objective function (F.1) can be changed slightly without destroying the optimality of $\hat{x}$. We need this in order to obtain strict inequalities in (1). Since the set M of solutions of the system of linear equations (F.2) and linear inequalities (FF.3) is bounded by the constraints (FF.3), this set is a polytope, and it follows from the theory of linear programming that the minimum of any linear objective function over M is attained at a vertex of the polytope (that is, at a basic solution of the linear program). Hence, $\hat{x}$ is a vertex of M. Since M has only finitely many vertices, we can define δ to be the minimum difference in objective function value (F.1) between $\hat{x}$ and any other vertex. If we replace d_{ij} by

$$d''_{ij} = \begin{cases} d_{ij} - \delta/n^2 & \text{if } \hat{x}_{ij} = 0, \quad \text{and} \\ d_{ij} & \text{if } \hat{x}_{ij} = 1, \end{cases}$$

we get a new objective function

$$\text{(F.1}'') \quad \text{minimize} \sum_{\{p_i, p_j\} \in G} x_{ij} d''_{ij}.$$

Now, we have

$$\sum_{\{p_i, p_j\} \in G} x_{ij} d''_{ij} = \sum_{\{p_i, p_j\} \in G} x_{ij} d_{ij} - \sum_{\hat{x}_{ij}=0} \frac{\delta}{n^2}$$

$$\geqslant \sum_{\{p_i, p_j\} \in G} x_{ij} d_{ij} - \binom{n}{2} \frac{\delta}{n^2} > \sum_{\{p_i, p_j\} \in G} x_{ij} d_{ij} - \tfrac{1}{2}\delta.$$

Thus, (F.1″) differs from (F.1) by at most $\frac{1}{2}\delta$, for all $x \in M$. This implies that $\hat{x}$ is still optimal with respect to (F.1″) among all vertices of M, and hence it is an optimal solution of the linear program (F.1″), (F.2), and (FF.3).

It remains to find radii r_i that realize the f-factor $\mathscr{F}$ corresponding to $\hat{x}$. Let us consider the dual program of (F.1″), (F.2), and (FF.3):

$$\text{maximize} \sum_{i=1}^{n} r_i f_i - \sum_{\{p_i, p_j\} \in G} \delta_{ij}$$

$$\text{subject to } -\delta_{ij} + r_i + r_j \leqslant d''_{ij} \quad \text{for } \{p_i, p_j\} \in G,$$

$$\text{with } \delta_{ij} \geqslant 0 \text{ and } r_i \text{ arbitrary.}$$

Here, the r_i are the dual variables associated with the vertices, and the δ_{ij} are associated with the constraints $x_{ij} \leqslant 1$. Complementarity implies that for $\hat{x}$ to be optimal, there must exist values r_i and δ_{ij} such that the following complementary slackness conditions hold:

$$\hat{x}_{ij} > 0 \text{ implies } r_i + r_j - \delta_{ij} = d''_{ij}, \quad \text{and}$$

$$\hat{x}_{ij} < 1 \text{ implies } \delta_{ij} = 0.$$

This means

$$\text{(I.1)} \quad \begin{cases} \hat{x}_{ij} > 0 \text{ implies } r_i + r_j \geqslant d''_{ij} = d_{ij}, \quad \text{and} \\ \hat{x}_{ij} < 1 \text{ implies } r_i + r_j \leqslant d''_{ij} = d_{ij} - \delta/n^2 < d_{ij}. \end{cases}$$

Thus, the f-factor $\mathcal{F}$ corresponding to $\hat{x}$ is realized by the radii r_i, which are just the dual variables of the linear program. □

Corollary 2.2. *A realizable f-factor of a graph is the unique optimal f-factor.*

Remark. The proof of Theorem 2.1 does not use the triangle inequality, and therefore Theorem 2.1 and Corollary 2.2 hold for arbitrary weighted graphs.

2.2. Consequences for necklace-trous

To apply the preceding results to necklace-tours, we have to state the following theorem which holds only for m-factors, that is, for f-factors where the degrees of all vertices are equal to the same integer m.

Theorem 2.3. *An m-factor is realizable if and only if it is realizable by positive radii.*

The proof of Theorem 2.3 is rather technical and will be given in Section 4. Since necklace-tours are tours realizable by positive radii and since tours are 2-factors, we can apply Theorem 2.3 and Corollary 2.2 to get the following result.

Corollary 2.4. *A necklace-tour of a set of points P is unique and it is the unique optimal 2-factor of P.*

Although this corollary suggests that the two-step procedure of constructing an optimal 2-factor and then testing its realizability is a viable way to test the necklace condition, we can exploit the realizability requirement already in the construction of the optimal 2-factor and reduce this to an optimal 2-factor problem in a bipartite graph. This problem is much simpler to solve than in the non-bipartite case although the algorithm we give is of the same asymptotic complexity as the one for the non-bipartite case (see Section 6).

2.3. Sparse supergraphs of m-factors

The following theorem is important because in Section 3 we will show that $\mathcal{G}^{(m)}$ of a set of points is a sparse graph.

Theorem 2.5. *A realizable m-factor $\mathcal{F}$ of a graph $\mathcal{G}$ is a subgraph of $\mathcal{G}^{(m)}$.*

Proof. Let $r_1, r_2, \ldots, r_n$ be radii that realize $\mathcal{F}$. According to Theorem 2.3, we can assume that they are positive. Now, we define a new set of radii as follows:

$$r_i' := \min\{r_i, d^{(m)}(p_i)\}.$$

The graph $\mathcal{F}'$ realized by the r_i' is a subgraph of $\mathcal{F}$ since $r_i' \leq r_i$. Nevertheless, every point for which the radius has decreased has still at least m neighbors in $\mathcal{F}'$: if

$r'_i < r_i$, then $r'_i = d^{(m)}(p_i)$, and $d^{(m)}(p_i) \geq d_{ij}$ for at least m points p_j. Therefore,

$$r'_i + r'_j > r'_i = d^{(m)}(p_i) \geq d_{ij}$$

for at least m points p_j.

Thus, $\mathcal{F}$ must be equal to $\mathcal{F}'$. On the other hand, $\mathcal{F}'$ is a subgraph of $G^{(m)}$ since $r'_i \leq d^{(m)}(p_i)$. □

Theorem 2.5 shows that a necklace-tour must have its edges in $\mathcal{G}^{(2)}$, and, by Corollary 2.2, it is the optimal 2-factor of $\mathcal{G}^{(2)}$.

Remark. The example shown in Fig. 2 demonstrates that Theorem 2.5 is best possible in the sense that a realizable 2-factor of P is not necessarily a subgraph of $\mathcal{G}^{(1)}$ of P: points 1 and 2 are not adjacent in $\mathcal{G}^{(1)}$ but they are in the necklace-tour. Similar examples exist for arbitrary positive integers m.

Remark. The example shown in Fig. 3 demonstrates that graph $\mathcal{G}^{(2)}$ of a finite set of points P does not always have a 2-factor because there is no cycle through point 1.

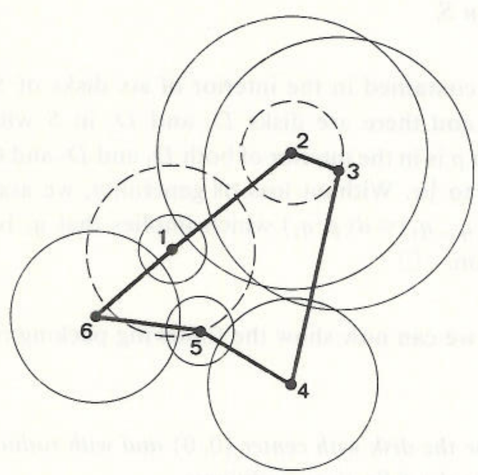

Fig. 2. Another necklace-tour.

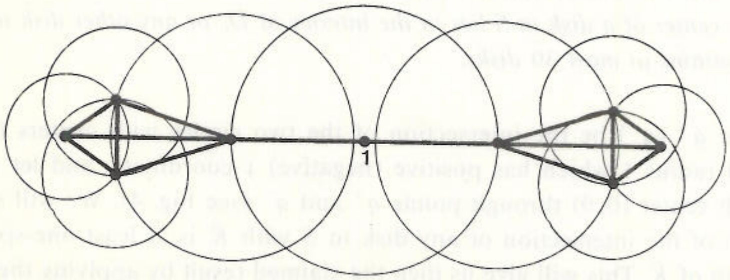

Fig. 3. A point set without realizable 2-factor.

Notice that Theorem 2.5 now implies that there are point sets without realizable 2-factor. The above type of argument can be used to show that, for every positive integer m, there are point sets without realizable m-factor. It is also true that an example can be constructed such that $\mathscr{G}^{(2)}$ has an optimal 2-factor which is a tour but which is not the optimal tour of $\mathscr{G}$.

3. $\mathscr{G}^{(m)}$ of a set of points in the plane is sparse

This section shows that, for each positive integer m, the graph $\mathscr{G}^{(m)}$ of a set of n points in the plane has at most $(31m-1)n$ edges. A linear upper bound on the number of edges of $\mathscr{G}^{(1)}$ can also be found in [1]. In the following proofs, $\angle(p_1, p_2, p_3)$ denotes the angle enclosed by the line segment connecting points p_2 and p_1 and the line segment connecting p_2 and p_3.

Lemma 3.1. *Let S be a set of closed disks such that no disk contains the center of another disk in S in its interior. Then any point p in the plane is contained in the interior of at most five disks in S.*

Proof. Suppose p is contained in the interior of six disks of S. Then p cannot be the center of a disk and there are disks D_1 and D_2 in S with centers q_1 and q_2 respectively, such that p is in the interior of both D_1 and D_2 and the angle $\angle(q_1, p, q_2)$ is less than or equal to $\frac{1}{3}\pi$. Without loss of generality, we assume that $d(p, q_1) \geqslant d(p, q_2)$. But now $d(q_2, q_1) \leqslant d(p, q_1)$ which implies that q_2 is also in the interior of D_1—a contradiction. □

Using Lemma 3.1, we can now show the following packing result for disks in the plane.

Lemma 3.2. *Let D_0 be the disk with center $(0,0)$ and with radius 1 and let S be a set of disks which satisfies the following conditions:*

(i) *all disks in S intersect D_0,*
(ii) *all disks in S have radius at least 1, and*
(iii) *no center of a disk in S lies in the interior of D_0 or any other disk in S.*

Then S contains at most 30 disks.

Proof. Let q^+ (q^-) be the intersection of the two circles with centers $(1,0)$ and $(2,0)$ and radius 1 which has positive (negative) y-coordinate, and let $\bar{K}$ be the circle with center $(0,0)$ through points q^+ and q^- (see Fig. 4). We will show that the length of the intersection of any disk in S with $\bar{K}$ is at least one-sixth of the total length of $\bar{K}$. This will give us then the claimed result by applying the previous lemma.

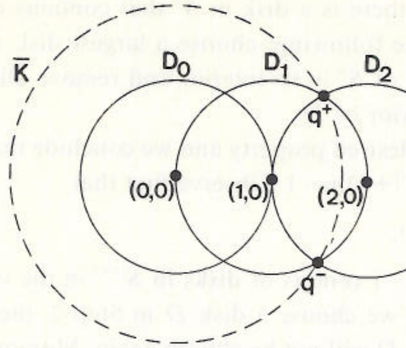

Fig. 4. Illustration of the proof of Lemma 3.2.

Observe first that both D_1, the disk with center $(1, 0)$ and radius 1, and D_2, the disk with center $(2, 0)$ and radius 1, contain one-sixth of $\bar{K}$. This is because the angle $\angle(q^+, (0,0), q^-)$ is exactly $\frac{1}{3}\pi$.

Now, consider a disk D in S with center at distance at least 2 from $(0, 0)$. Without loss of generality, we assume that the center of D is on the positive x-axis. Then D contains D_2, since it intersects D_0 by assumption, which implies that it contains at least one-sixth of $\bar{K}$. If the distance of the center of D from $(0, 0)$ is between 1 and 2, including the limits, and if it lies on the positive x-axis, then D contains the intersection of D_2 and $\bar{K}$. Since no disk in S has distance smaller than 1 from $(0, 0)$, we conclude that every disk in S intersects $\bar{K}$ in an arc of length at least one-sixth of the total length of $\bar{K}$.

Now, we choose six points on $\bar{K}$ which are the vertices of a regular hexagon such that no point lies on the boundary of a disk in S. We know that each of these points lies in at most five disks in S and every disk in S contains at least one of these points. Consequently, $|S| \leq 5 \cdot 6 = 30$. □

Using Lemma 3.2, we prove the main result of this section which guarantees the sparsity of $\mathscr{G}^{(m)}$.

Theorem 3.3. *Let P be a set of n points in the plane and let m be a positive integer number. Then the edge set $G^{(m)}$ of the graph $\mathscr{G}^{(m)}$ of P contains at most $(31m-1)n$ edges.*

Proof. We claim that a smallest disk D_0 in $S^{(m)} = \{D^{(m)}(p) \mid p \in P\}$ intersects at most $31m-1$ disks in $S^{(m)}$. From this the theorem follows by a simple inductive argument.

Let S be the set of disks in $S^{(m)}$ that intersect D_0. We reduce S to a subset S'' of S such that no center of a disk in S'' lies in the interior of D_0 or of any other disk in S''. The reduction is done by the following procedure.

Step 1: Remove all disks from S that have their centers in the interior of D_0. Let S' be the resulting set.

Step 2: $S'' := S'$. While there is a disk in S'' that contains centers of other disks in S'' in its interior, do the following: choose a largest disk D in S'' that contains the center of another disk of S'' in its interior and remove all disks from S'' whose centers belong to the interior of D.

Apparently, S'' has the desired property and we conclude that $|S''| \leq 30$. It remains to be shown that $|S| \leq (|S''|+1)m-1$. Observe first that

$$|S'| \geq |S|-(m-1), \tag{$*$}$$

since there are at most $m-1$ centers of disks in $S^{(m)}$ in the interior of D_0.

Analogously, whenever we choose a disk D in Step 2, then we remove at most $m-1$ disks from $S^{(m)}$ and D will not be chosen again. Moreover, the special choice of D guarantees that D will not be removed from S'' in a later step. Consequently, we remove at most $m-1$ disks for each disk which remains in S'' which implies

$$|S''| \geq |S'|/m. \tag{$**$}$$

From ($*$) and ($**$) it follows that

$$|S| \leq |S'|+m-1 \leq m|S''|+m-1 = (|S''|+1)m-1 \leq 31m-1$$

which proves the theorem. □

At this point we would like to thank David Avis who brought to our attention that improvements of Lemma 3.2 have been proven in [24] and, independently, in [3]. Both papers show that a disk that satisfies the conditions of Lemma 3.2 intersects at most 18 other disks which is best possible. Their proofs are considerably more involved than the one presented to prove the constant 30. Using their result, Theorem 3.3 can be improved to $(19m-1)n$. This bound is still a factor of 2 off the conjectured tight bound which is $9n+o(n)$ for $m=1$.

4. Reduction of the system of inequalities

In this section, we show that positive radii are always sufficient to realize m-factors, if they are realizable at all (see Theorem 2.3). This will allow us to eliminate many of the inequalities in system (1). More specifically, we show that the following systems of linear inequalities are equivalent with respect to their solvability. Let $\mathscr{F}$ be an m-factor of $\mathscr{G}$. The two systems of inequalities are

$$(2) \quad \begin{cases} r_i + r_j \geq d_{ij} & \text{for } \{p_i, p_j\} \in F, \quad \text{and} \\ r_i + r_j < d_{ij} & \text{for } \{p_i, p_j\} \in G^{(m)} - F, \end{cases}$$

and

$$(3) \quad \begin{cases} r_i + r_j \geq d_{ij} & \text{for } \{p_i, p_j\} \in F, \\ r_i + r_j < d_{ij} & \text{for } \{p_i, p_j\} \in G - F, \quad \text{and} \\ r_i > 0 & \text{for } p_i \in P. \end{cases}$$

Notice that (2) has considerably fewer inequalities than (1) and (3) if $\mathscr{G}^{(m)}$ is sparse.

Theorem 4.1. *The following statements are equivalent*:

(i) (1) *is solvable* (*that is*, $\mathcal{F}$ *is realizable*),

(ii) (2) *is solvable, and*

(iii) (3) *is solvable* (*that is*, $\mathcal{F}$ *is realizable by positive radii*).

The proof of Theorem 4.1 is going to be rather technical and involved. A shorter and more intuitive proof would certainly be desirable.

Proof. The conclusions from (iii) to (i) and from (i) to (ii) are immediate. To conclude from (ii) to (iii) we proceed in several steps. First, we need two technical lemmas.

Lemma 4.2. *Let* r_i *be radii fulfilling* (2) *for an m-factor* $\mathcal{F}$. *If* $r_i \leqslant 0$ *and* $r_l \geqslant d_{il} \leqslant d^{(m)}(p_l)$, *then* $\{p_i, p_l\} \in \mathcal{F}$.

Proof. Assume that there are points p_i with nonpositive r_i, $r_l \geqslant d_{il} \leqslant d^{(m)}(p_l)$ and $\{p_i, p_l\} \notin F$, and let p_i be the point with smallest radius r_i among these. Let N be the set of the m neighbors of p_i in $\mathcal{F}$. We have

$$r_j + r_i \geqslant d_{ij} \quad \text{for } p_j \in N, \tag{$*$}$$

and hence $r_j \geqslant d_{ij}$ for all $p_j \in N$.

Next, we exclude the possibility that the distance between p_j and p_i is less than or equal to $d^{(m)}(p_j)$ for each point p_j in N. Assume we have $d_{ij} \leqslant d^{(m)}(p_j)$ for all $p_j \in N$. Then we get

$$d^{(m)}(p_j) + d^{(m)}(p_{j'}) \geqslant d_{ij} + d_{ij'} \geqslant d_{jj'},$$

for $p_j, p_{j'} \in N \cup \{p_i\}$, and hence $\{p_j, p_{j'}\} \in G^{(m)}$ for all $p_j, p_{j'} \in N \cup \{p_l\}$. Furthermore,

$$r_j + r_{j'} \geqslant d_{ij} + d_{ij'} \geqslant d_{jj'},$$

for all $p_j, p_{j'} \in N \cup \{p_l\}$, by the triangle inequality. This implies $\{p_j, p_{j'}\} \in F$ since, otherwise, this contradicts $r_j + r_{j'} < d_{jj'}$ in (2). Consequently, every $p_j \in N$ is adjacent to every other $p_{j'} \in N \cup \{p_l\}$ and to p_i. Point p_j therefore has degree at least $m+1$ in $\mathcal{F}$—a contradiction.

Thus, we proved that there is a point p_j with $\{p_j, p_i\} \in F$ and $d_{ij} > d^{(m)}(p_j)$. The value of $d^{(m)}(p_j)$ exceeds d_{jk} for at least m points p_k, hence we have

$$d_{jk} \leqslant d^{(m)}(p_j) < d_{ji}, \tag{$**$}$$

which implies that $\{p_j, p_k\} \in G^{(m)}$ for at least m points p_k, $k \neq j$ and $k \neq i$. Since p_j is adjacent to only m neighbors in $\mathcal{F}$ and one of them is p_i, one of the points p_k must not be adjacent to p_j in $\mathcal{F}$. For this point p_k, we then have $r_j + r_k < d_{jk}$ from (2), because $\{p_j, p_k\} \in G^{(m)}$. On the other hand, $r_j + r_i \geqslant d_{ij} > d_{jk}$ from ($*$) and ($**$), which implies $r_k < r_i \leqslant 0$. In addition, it implies $r_j \geqslant d_{jk}$, and we have $d_{jk} \leqslant d^{(m)}(p_j)$ from ($**$). Since we also have $\{p_j, p_k\} \notin F$, the existence of point p_k with radius $r_k < r_i$ (together with p_j) contradicts the minimality of r_i. $\square$

Lemma 4.3. *If a set of values r_i, $i = 1, 2, \ldots, n$, fulfills* (2), *then*

$$r_i' := \begin{cases} d^{(m)}(p_i) & \text{if } r_i > d^{(m)}(p_i), \\ 0 & \text{if } r_i < 0, \text{ and} \\ r_i & \text{otherwise,} \end{cases}$$

also fulfill (2).

Proof. We now have $0 \leq r_i' \leq d^{(m)}(p_i)$ for all i. In the proof below, we distinguish two cases, namely the case that an edge belongs to F and the case that it belongs to $G^{(m)} - F$. We treat the second case first.

(i) If $\{p_i, p_j\} \in G^{(m)} - F$, then we have $r_i + r_j < d_{ij}$. The corresponding inequality $r_i' + r_j' < d_{ij}$ is violated only if one variable (say, r_i) is negative which implies $r_i' = 0$. In this case, $r_j \geq d_{ij}$ implies $r_j \geq r_j' \geq d_{ij}$ and $d^{(m)}(p_j) \geq d_{ij}$, which contradicts Lemma 4.2. Thus, we have

$$r_j' < d_{ij} \quad \text{and} \quad r_i' + r_j' = r_j' < d_{ij}.$$

(ii) This part of the proof is an extension of the proof of Theorem 2.5. If $\{p_i, p_j\} \in F$, then we have $r_i + r_j \geq d_{ij}$. The corresponding inequality $r_i' + r_j' \geq d_{ij}$ is violated only if one variable r_i or r_j (say, r_i) is greater than $d^{(m)}(p_i)$ which implies $r_i' = d^{(m)}(p_i)$. We have $d^{(m)}(p_i) \geq d_{ik}$ for at least m points p_k. Hence,

$$r_i' + r_k' \geq r_i' = d^{(m)}(p_i) \geq d_{ik} \quad \text{and} \quad d^{(m)}(p_i) + d^{(m)}(p_k) \geq d^{(m)}(p_i) \geq d_{ik},$$

which implies $\{p_i, p_k\} \in G^{(m)}$ for at least m points p_k. We have $\{p_i, p_k\} \in F$ for each of these points, since, otherwise, $r_i' + r_k' < d_{ik}$, by part (i) of this proof. But there are only m points p_k with $\{p_i, p_k\} \in F$. Thus $r_i' + r_k' \geq d_{ik}$ holds for all points p_k with $\{p_i, p_k\} \in F$ and therefore also for point p_j. □

Now, we come to the conclusion of the proof of Theorem 4.1. By the preceding lemma, we have transformed a solution of (2) into a solution fulfilling the additional constraints $0 \leq r_i \leq d^{(m)}(p_i)$. Now, for $\{p_i, p_j\} \notin G^{(m)}$, we get the remaining inequalities

$$r_i + r_j \leq d^{(m)}(p_i) + d^{(m)}(p_j) < d_{ij}$$

for free. Let $\delta = \min\{d_{ij} - r_i - r_j \mid \{p_i, p_j\} \in G - G^{(m)}\} > 0$. Then $r_i'' := r_i + \frac{1}{3}\delta$ still fulfills all inequalities and, in addition, we have $r_i'' > 0$. □

5. Testing the realizability of a given *f*-factor

In order to check whether or not a given *f*-factor is realizable, we have to check whether or not the system of linear inequalities (1) has a solution. Section 5.1 deals with this problem. In the case of *m*-factors, the number of inequalities can be reduced from $\binom{n}{2}$ to $O(mn)$ by Theorem 4.1 (see system (2)). In order to apply this reduction, we must know the graph $\mathscr{G}^{(m)}$. The construction of $\mathscr{G}^{(m)}$ for a set of points in the plane is the subject of Section 5.2. Finally, as a byproduct of the results in Section 5.1, we prove an interesting alternating path characterization of realizable f-factors.

5.1. *Testing the feasibility of the linear program* (2) *or* (1)

We want to check the feasibility, that is, the existence of a solution, of a system of inequalities of the form

$$(4)\qquad \begin{cases} r_i + r_j \geq d_{ij} & \text{for } \{p_i, p_j\} \in F, \quad \text{and} \\ r_i + r_j < d_{ij} & \text{for } \{p_i, p_j\} \in \bar{F}, \end{cases}$$

where $\bar{F}$ is either $G^{(m)} - F$ or $G - F$ depending on the case we consider. In case of solvability, we also want to find a solution.

To test the feasibility of a sysstem of linear inequalities, several procedures have been proposed in the literature ranging from pairwise elimination schemes for general systems suggested by Fourier [11] (see also [8]) to polynomial algorithms for systems in which at most two variables occur in each inequality (see [19] which is based on a technique in [29]). In particular, Megiddo [19] gives an $O(mn^3 \log m)$ time algorithm to solve systems of m inequalities and n variables, where each inequality contains at most two variables.

In our case, however, the special situation that all coefficients are equal to +1 or −1 allows us to impose a directed graph structure on the problem and to utilize this structure for a more efficient solution than the one given in [19] for the more general case. Our technique also follows the ideas in Shostak [29].

For the sake of symmetry, we double the number of inequalities by rewriting system (4) as follows:

$$(4')\qquad \begin{cases} -r_i \leq r_j - d_{ij} & \text{for } \{p_i, p_j\} \in F, \\ -r_j \leq r_i - d_{ij} & \text{for } \{p_i, p_j\} \in F, \\ r_i < -r_j + d_{ij} & \text{for } \{p_i, p_j\} \in \bar{F}, \quad \text{and} \\ r_j < -r_i + d_{ij} & \text{for } \{p_i, p_j\} \in \bar{F}. \end{cases}$$

In order to set up our directed graph, it is advantageous to introduce a new variable $\bar{r}_i$ for each r_i, where $\bar{r}_i$ represents $-r_i$. We denote by V the set $\{r_i \mid 1 \leq i \leq n\}$ and by $\bar{V}$ the set $\{\bar{r}_i \mid 1 \leq i \leq n\}$. Using the new variables, we rewrite system (4′) to

$$(5a)\qquad \begin{cases} \bar{r}_i \leq r_j - d_{ij} & \text{for } \{p_i, p_j\} \in F, \\ \bar{r}_j \leq r_i - d_{ij} & \text{for } \{p_i, p_j\} \in F, \\ r_i < \bar{r}_j + d_{ij} & \text{for } \{p_i, p_j\} \in \bar{F}, \quad \text{and} \\ r_j < \bar{r}_i + d_{ij} & \text{for } \{p_i, p_j\} \in \bar{F}. \end{cases}$$

With the additional constraints

$$(5b)\qquad r_i = -\bar{r}_i \quad \text{for } r_i \in V,$$

system (5) consisting of all constraints in (5a) and (5b) is equivalent to system (4′) and therefore to system (4).

All inequalities in (5a) are now of the form

$$(6) \quad x \leqslant y + c_{xy} \quad \text{with } x \in \bar{V}, y \in V, \text{ and } c_{xy} \text{ a real number, or}$$

$$(6') \quad x < y + c_{xy} \quad \text{with } x \in V, y \in \bar{V}, \text{ and } c_{xy} \text{ a real number.}$$

We construct a directed graph $\mathcal{G}_0$ with node set $V_0 = V \cup \bar{V} \cup \{s\}$, where s is a new node not in $V \cup \bar{V}$. For each inequality of system (5a), when it is written in the form (6) or (6′), we have an arc from x to y with weight c_{xy}. Moreover, we add arcs with weight zero from s to every node in $V \cup \bar{V}$. Following an earlier convention, we let G_0 denote the arc set of $\mathcal{G}_0$.

Lemma 5.1. *If $\mathcal{G}_0$ has a directed cycle of weight at most zero, then* (2) *has no solution.*

Proof. Consider a directed path $(x_0, x_1, \ldots, x_k)$, $k \geqslant 2$, in $\mathcal{G}_0$ that does not contain node s. It can be easily seen that the nodes x_i, $0 \leqslant i \leqslant k$ are alternately from V and $\bar{V}$, that is, the inequalities corresponding to the arcs in this path are alternately of type (6) and of type (6′). Combining the inequalities corresponding to these arcs results in the inequality

$$x_0 < x_k + c_{x_0x_1} + c_{x_1x_2} + \cdots + c_{x_{k-1}x_k}.$$

Thus, if $x_0 = x_k$ and the sum of the arc weights is nonpositive, then this reveals the infeasibility of system (5a). Note that node s does not have any incoming arc which implies that no directed cycle contains s. Since the system (5a) is weaker than (2), this contradiction implies that (2) has no solution. □

Unless system (2) is infeasible, graph $\mathcal{G}_0$ has no nonpositive and therefore no negative directed cycle, by Lemma 5.1. For each node z of $\mathcal{G}_0$, the weight of a minimum weight path from s to z is therefore well-defined; we let $-l_z$ denote this weight. Let (x, y) be an arc in G_0. Since the minimum weight of any path from s to y cannot be greater than the minimum weight of any path from s to x plus the arc weight from x to y, we have $-l_y \leqslant -l_x + c_{xy}$, and therefore $l_x \leqslant l_y + c_{xy}$ for each inequality $x \leqslant y + c_{xy}$ or $x < y + c_{xy}$ in (5a).

Notice that $z \in V \cup \bar{V}$ is a node of graph $\mathcal{G}_0$ and a variable in (5a) at the same time. It therefore makes sense to set $z := l_z$ for each $z \in V \cup \bar{V}$. This assignment satisfies the system of inequalities which is obtained from (5a) by weakening each strict inequality of type (6′) to a nonstrict inequality. To obtain a solution for system (5a), we modify the above assignment locally. This is described in the next paragraph.

Let $\hat{G}_0$ be the set of arcs (x, y) with $l_x = l_y + c_{xy}$, and let $\hat{G}_0$ be the directed graph with node set V_0 and arc set $\hat{G}_0$. Note that a cycle in $\hat{G}_0$ corresponds to a cycle of weight zero in $\mathcal{G}_0$. By Lemma 5.1, $\hat{\mathcal{G}}_0$ has a directed cycle only if system (2) is infeasible. Hence, we can assume that $\mathcal{G}_0$ has only positive cycles. In this case, we can relabel the elements $z_1, z_2, \ldots, z_{2n}$ of $V \cup \bar{V}$ in such a way that $(z_i, z_j) \in \hat{G}_0$

implies $i<j$. (The ordering is a topological sort of $V \cup \bar{V}$ with respect to $\hat{G}_0$.) Let δ be the minimum of $l_y + c_{xy} - l_x$ for all arcs (x, y) in $G_0 - \hat{G}_0$. By definition of $\hat{G}_0$, $\delta > 0$. In the following lemma we refer to this real number δ and to an enumeration of the elements of $V \cup \bar{V}$ as described above.

Lemma 5.2. *If $\mathscr{G}_0$ has only positive cycles, then the assignment*

$$z_i := l'_{z_i} = l_{z_i} + \frac{i-1}{2n}\,\delta, \quad 1 \leqslant i \leqslant 2n,$$

satisfies (5a).

Proof. Consider an arc (z_i, z_j) in G_0. We know that $l_{z_i} \leqslant l_{z_j} + c_{z_i z_j}$. If $i<j$, then

$$l'_{z_i} = l_{z_i} + \frac{i-1}{2n}\,\delta < l_{z_j} + \frac{j-1}{2n}\,\delta + c_{z_i z_j} = l'_{z_j} + c_{z_i z_j}$$

holds, because $(i-1)\delta/2n < (j-1)\delta/2n$. If $i>j$, then (z_i, z_j) is not an arc of $\hat{\mathscr{G}}_0$ and so $l_{z_j} + c_{z_i z_j} - l_{z_i} \geqslant \delta$ which implies

$$l'_{z_j} + c_{z_i z_j} - l'_{z_i} = l_{z_j} + \frac{j-1}{2n}\,\delta + c_{z_i z_j} - l_{z_i} - \frac{i-1}{2n}\,\delta \geqslant \delta + \frac{j-i}{2n}\,\delta = \frac{\delta(2n+j-i)}{2n} > 0.$$

Hence, $l'_{z_i} < l'_{z_j} + c_{z_i z_j}$. Thus all inequalities in (5a) are fulfilled (even if we strengthen nonstrict inequalities to strict ones). □

By the preceding two lemmas, we can either detect the infeasibility of system (4) or give a solution for system (5a). The original system (4) is equivalent to system (5) consisting of all constraints in (5a) and (5b). Below, we show how a solution for (5a) can be used to obtain a solution for (5). To help the notation, we write l'_{r_i} and $l'_{\bar{r}_i}$ for the values of the assignment described above. To avoid confusion, we note that r_i is in general not the same as z_i. However, for each $1 \leqslant i \leqslant n$, there are indices j and j' such that $r_i = z_j$ and $\bar{r}_i = z_{j'}$, and the values of j and j' are determined by the topological sort used to relabel the elements of $V \cup \bar{V}$.

Lemma 5.3. *The assignment*

$$r_i := \tfrac{1}{2}(l'_{r_i} - l'_{\bar{r}_i}) \quad \textit{for } r_i \in V, \quad \textit{and}$$

$$\bar{r}_i := \tfrac{1}{2}(l'_{\bar{r}_i} - l'_{r_i}) \quad \textit{for } \bar{r}_i \in \bar{V}$$

fulfills (5a) *and* (5b).

Proof. Consider an inequality of the form $r_i < \bar{r}_j + d_{ij}$ in (5a). Then there is also the inequality $r_j < \bar{r}_i + d_{ij}$ in (5a). Hence, we have $l'_{r_i} < l'_{\bar{r}_j} + d_{ij}$ whenever we have $l'_{r_j} < l'_{\bar{r}_i} + d_{ij}$. Adding these inequalities, we get

$$\tfrac{1}{2}(l'_{r_i} - l'_{\bar{r}_i}) < \tfrac{1}{2}(l'_{\bar{r}_j} - l'_{r_j}) + d_{ij}$$

which shows that the assignment defined in Lemma 5.3 satisfies $r_i < \bar{r}_j + d_{ij}$. We proceed analogously for inequalities of the form $\bar{r}_i \leq r_j - d_{ij}$ in (5a). Conditions (5b) are trivially satisfied which completes the proof of the lemma. □

The following theorem summarizes the results of Lemmas 5.1 through 5.3.

Theorem 5.4. *An f-factor is realizable (equivalently, system* (1) *has a solution) if and only if $\mathcal{G}_0$ has only positive cycles. In this case,* $r_i := \frac{1}{2}(l'_{r_i} - l'_{\bar{r}_i})$ *are radii that realize $\mathcal{F}$.*

Using Theorem 4.1 and Lemma 4.3, we further get the following result which is concerned with m-factors.

Theorem 5.5. *An m-factor $\mathcal{F}$ is realizable (equivalently, system* (2) *has a solution) if and only if $\mathcal{G}_0$, defined for the edges of $\mathcal{G}^{(m)}$, has only positive cycles. In this case,*

$$r_i := \max\{0, \min\{d^{(m)}(p_i), \tfrac{1}{2}(l'_{r_i} - l'_{\bar{r}_i})\}\}$$

are (nonnegative) radii that realize $\mathcal{F}$.

Finally, we address the algorithmic issues raised by the above results.

Theorem 5.6. *The feasibility of a system of inequalities of the form* (4) *can be tested in* $O(n(|F|+|\bar{F}|))$ *time and* $O(|F|+|\bar{F}|)$ *space, where n is the number of variables. If a solution exists, it can be found within the same time and space bounds.*

Proof. We construct the graph $\mathcal{G}_0$ in $O(|F|+|\bar{F}|)$ time and space from the system (4). Apparently, $\mathcal{G}_0$ has $2n+1$ nodes and $O(|F|+|\bar{F}|)$ edges. The single source shortest path problem for node s can be solved by the Bellman–Ford algorithm in $O(n(|F|+|\bar{F}|))$ time and $O(|F|+|\bar{F}|)$ space (see [16]). Now, if either a negative cycle has been detected or $\hat{G}_0$ has a cycle, then the system is not solvable; otherwise, it is.

A topological sort of $\hat{G}_0$ can be performed in $O(n)$ time and space (see [20]). Finally, the remaining calculations following the lines of Lemmas 5.2 and 5.3 which are necessary to obtain the realizing radii can be easily done within the required time and space bounds. □

When we test an f-factor for realizability we have $F \cup \bar{F} = G$ in (4) which implies the following result.

Corollary 5.7. *An f-factor can be tested for realizability in* $O(n^3)$ *time and* $O(n^2)$ *space, and if it is realizable, realizing radii can be found within the same time and space bounds.*

More specifically, given a weighted graph $\mathcal{G}$ with n vertices fulfilling the triangle inequality and an m-factor $\mathcal{F}$, we can test in $O(n|G^{(m)}|)$ time and $O(|G^{(m)}|)$ space (excluding the space needed to store $\mathcal{G}$) whether $\mathcal{F}$ can be realized. If it can be realized, then realizing radii can be found within the same bounds. If $\mathcal{G}$ is the complete distance graph of a set of n points in the plane, then the sparsity results of Section 3 imply a faster algorithm than for arbitrary f-factors.

Corollary 5.8. *Given the graph $\mathcal{G}^{(m)}$, an m-factor for n points in the plane can be tested for realizability in $O(mn^2)$ time and $O(mn)$ space, and if it is realizable, realizing radii can be found within the same time and space bounds.*

5.2. Construction of $\mathcal{G}^{(m)}$ for points in the plane

By Theorem 3.3, we know that the size of $\mathcal{G}^{(m)}$ of n points in the plane is $O(mn)$. We will take advantage of this property to design an efficient algorithm for constructing $\mathcal{G}^{(m)}$.

Theorem 5.9. *The graph $\mathcal{G}^{(m)}$ of a set P of n points in the plane can be computed in $O(m^2 n \log n)$ time and $O(m^2 n)$ space.*

Proof. In order to compute $\mathcal{G}^{(m)}$ in $O(m^2 n \log n)$ time and $O(m^2 n)$ space, we first calculate the value $d^{(m)}(p)$ for each point p in P. To this end, we consider the partition of the plane, where each region is the locus of points which are closer to some m points in P than to any other point in P. This partition is known as the so-called mth-order Voronoi diagram of P (see [23]), and can be constructed in $O(m^2 n \log n)$ time and $O(m^2 n)$ space (see [17]). Next, we compute the mth-nearest neighbor of each of the n points in P in $O(mn \log n)$ time. This is done by locating for each point the region of the diagram it lies in (for a solution to this point location problem, see [9]), and by determining the furthest of the m nearest points given by the region located. Thus, in $O(m^2 n \log n)$ time, we can determine the real numbers $d^{(m)}(p)$ for all points p in P, and therefore also the set of disks $S = \{D^{(m)}(p) \mid p \in P\}$.

Next, we find all pairs $\{p_1, p_2\}$ of points in P for which $D^{(m)}(p_1) \cap D^{(m)}(p_2) \neq \emptyset$. By construction of the disks $D^{(m)}(p)$, $p \in P$, no disk in S is contained in the interior of another disk in S. Hence, $D^{(m)}(p_1) \cap D^{(m)}(p_2) \neq \emptyset$ if and only if the bounding circles intersect. We thus have the problem of reporting all intersecting pairs of n circles. Using a technique of [4], this can be done in $O(n \log n + k \log n)$ time, where k is the number of reported pairs. By Theorem 3.3, we have $k = O(mn)$ which implies that the overall complexity is in $O(m^2 n \log n)$. For all algorithms mentioned, $O(m^2 n)$ space is sufficient which concludes the proof of the theorem. □

There is a straightforward algorithm for constructing $\mathcal{G}^{(m)}$ for n points in the plane that takes $O(n^2)$ time. The amount of space required by this trivial algorithm is $O(mn)$ which is better than the space complexity of the more sophisticated algorithm described in the proof of Theorem 5.9, at least for nonconstant values of

m. Still, the time needed by this trivial algorithm is dominated by the time complexities of the other steps of our overall algorithm (see Corollary 5.8), which makes it a desirable method because of its low space requirements. Nevertheless, Theorem 5.9 shows that the current bottle-neck of our algorithm consists of these other steps and that a considerably faster algorithm is obtained if one succeeds to speed up these other steps.

From Corollary 5.8, we get now the following result.

Theorem 5.10. *Given an m-factor $\mathscr{F}$ for n points in the plane, we can decide in* $O(mn^2)$ *time and* $O(mn)$ *space whether $\mathscr{F}$ is realizable. If $\mathscr{F}$ is realizable, then positive radii that realize $\mathscr{F}$ can be constructed within the same time and space bounds.*

5.3. An alternating cycle characterization of realizable f-factors

The results of Section 5.1 enable us to give a combinatorial characterization of realizable f-factors. It extends the well-known characterization of optimal f-factors by the non-existence of negative alternating cycles.

We begin by introducing a few definitions. A *cycle* of length $i \geqslant 2$ in a graph is a sequence of edges of the form $\{p_1, p_2\}, \{p_2, p_3\}, \ldots, \{p_i, p_1\}$. An *alternating cycle* $\mathscr{C}$ with respect to a given f-factor $\mathscr{F}$ is a cycle of even length such that exactly every other edge of $\mathscr{C}$ belongs to $\mathscr{F}$. The *weight* $w(\mathscr{C})$ of $\mathscr{C}$ is defined as follows:

$$w(\mathscr{C}) = \sum_{\{p_i, p_j\} \in C - F} d_{ij} - \sum_{\{p_i, p_j\} \in C \cap F} d_{ij}.$$

The following is a well-known characterization of optimal f-factors.

Theorem 5.11. *An f-factor is optimal if and only if there is no alternating cycle of negative weight that contains no edge twice (repetition of vertices is allowed).*

The theorem follows from the fact that any f-factor can be obtained from a given f-factor $\mathscr{F}$ by a sequence of exchanges of edges in $\mathscr{F}$ with edges not in $\mathscr{F}$ along pairwise disjoint alternating cycles without repeated edges (see [5, p. 147]). In the case of 1-factors (that is, complete matchings) the theorem can be strengthened to forbid also repetitions of vertices of alternating cycles.

Now, we adapt the characterization so that it holds for f-factors that are optimal and realizable.

Theorem 5.12. *An f-factor is realizable (and therefore optimal) if and only if there is no alternating cycle of non-positive weight, where the cycle is allowed to contain an edge more than once.*

Proof. According to the construction in Section 5.1, every alternating cycle in $\mathscr{G}$ with respect to $\mathscr{F}$ corresponds exactly to a cycle in $\mathscr{G}_0$, and vice versa. Moreover, the weight of the alternating cycle is equal to the weight of the corresponding cycle in $\mathscr{G}_0$. The theorem follows now from Theorem 5.4. □

6. Finding a realizable f-factor

By Theorem 2.1, we need only solve the fractional f-factor problem specified by (F.1), (F.2), and (FF.3) in order to find a realizable f-factor. Extending the reduction given in [6, 2] from the fractional f-factor problem to the assignment problem, we can reduce the fractional f-factor problem to an n times n capacitated transportation problem specified below. For consistency, we set $d_{ii} = \infty$ for $1 \leq i \leq n$.

$$\text{(C)} \quad \begin{cases} \text{minimize } \sum_{i=1}^{n} \sum_{j=1}^{n} y_{ij} d_{ij} \\ \text{subject to } \sum_{j=1}^{n} y_{ij} = f_i \quad \text{for } 1 \leq i \leq n, \\ \sum_{i=1}^{n} y_{ij} = f_j \quad \text{for } 1 \leq j \leq n, \quad \text{and} \\ 0 \leq y_{ij} \leq 1 \quad \text{for } 1 \leq i, j \leq n. \end{cases}$$

This is essentially the same f-factor problem but on a bipartite graph with twice as many vertices. The complementary slackness conditions for this problem and its dual are as follows:

$$\text{(I.2)} \quad \begin{cases} y_{ij} > 0 \text{ implies } u_i + v_j \geq d_{ij}, \quad \text{and} \\ y_{ij} < 1 \text{ implies } u_i + v_j \leq d_{ij}. \end{cases}$$

If a set of values x_{ij}, r_i fulfills (F.2), (FF.3), and (I.1), a pair of primal and dual solutions y_{ij} and u_i, v_j fulfilling (C) and (I.2) can be defined by setting

$$y_{ij} := y_{ji} := x_{ij} \quad \text{and} \quad u_i := v_i := r_i.$$

Conversely, from a solution fulfilling (C) and (I.2), we can compute a solution of (F.2), (FF.3), and (I.1) as follows:

$$x_{ij} := \tfrac{1}{2}(y_{ij} + y_{ji}) \quad \text{and} \quad r_i := \tfrac{1}{2}(u_i + v_i).$$

Since the values y_{ij} of an optimal basic solution of (C) are integral, it is necessary and sufficient for x_{ij} to be integral that $y_{ij} = y_{ji}$ for all i, j, that is, that the solution is symmetric.

By a straightforward extension of Theorem 2.1, we now get the following result.

Theorem 6.1. *A realizable f-factor exists if and only if the solution of the capacitated transportation problem* (C) *is symmetric and unique.*

In the cases where the realizable factor can be restricted to be contained in a subgraph, we need only consider the arcs corresponding to the edges of this subgraph. (There are two arcs (p_i, p_j) and (p_j, p_i) corresponding to each edge $\{p_i, p_j\}$ of the original subgraph.)

The remainder of this section discusses the complexity results that are obtained by a number of different ways to solve the transportation problem.

(A) If we use the shortest augmenting path method (see [10, 31]), we need at most $\sum_{i=1}^{n} f_i$ flow augmentations each involving a single source shortest path computation. For each shortest path computation, we can either use Dijkstra's original algorithm (see [7]), or the variation with priority queues implemented as ordinary heaps or as Fibonacci-heaps (see [12]), whichever is the simplest algorithm that achieves the desired time bound. The time-complexity of the three methods is respectively $O(n^2)$, $O(e \log n)$, and $O(e + n \log n)$, where e is the number of edges of the graph. The space requirement in all three cases is $O(e+n)$. This implies an $O((\sum f_i)n^2)$ time algorithm for finding arbitrary realizable f-factors and an $O(mn(mn + n \log n))$ time algorithm for finding realizable m-factors in the plane. The latter result follows from the above results since we have $\sum f_i = mn$ and $e = O(mn)$. Note that the complexity of $O((\sum f_i)n^2)$ for realizable f-factors is matched by an algorithm of [13] which always finds an optimal f-factor. For constant m, realizable m-factors can thus be found in $O(n^2 \log n)$ time which is again matched by an algorithm of [13] for computing optimal m-factors. Both algorithms of [13] are hard to implement.

(B) By using the algorithms of Orlin [21], we can reduce the time needed for the solution of the transportation problem to $O(n^3 \log(\max\{f_i \mid 1 \leq i \leq n\}))$ for general f-factors, and to $O(n^2 \log m(m + \log n))$, for m-factors in the plane. The first bound is incomparable to the time bound given in the preceding paragraph, and the second bound is asymptotically better than the time bound above if m gets larger.

Below, we summarize the complexity results obtained for finding realizable m-factors and realizable tours.

Theorem 6.2. *Given a set of n points in the plane, the necklace condition can be tested in* $O(n^2 \log n)$ *time and* $O(n)$ *space. If the necklace condition is satisfied, the necklace tour and a set of radii that realize it can be constructed within the same asymptotic time and space bounds.*

Theorem 6.3. *Given a set of n points in the plane, the existence of a realizable m-factor can be tested in* $O(n^2 \log m(m + \log n))$ *time and* $O(mn)$ *space. If a realizable m-factor exists, it can be constructed within the same asymptotic time and space bounds.*

7. Concluding remarks

We have presented an algorithm which tests the necklace condition for a finite point set. We have not attempted to extend all results to their outmost generality. For this reason, we mention a few immediate extensions of the techniques offered in this paper.

(1) All results described in this paper which are necessary for an extension of the main results to three and higher dimensions generalize nicely which implies that Theorems 6.2 and 6.3 also hold in higher dimensions where f-factors are realized by balls. It is not clear whether or not the sophisticated construction of $\mathscr{G}^{(2)}$

generalizes even to three dimensions, but for the generalization of Theorems 6.2 and 6.3, we can substitute it by straightforward methods without sacrificing the complexity bounds. It is true, however, that the constants in the complexity of the algorithm increase when the number of dimensions increases.

(2) All techniques of this paper generalize if we replace the Euclidean metric by other L_p-metrics, $p = 1, 2, \ldots, \infty$.

Preliminary computational tests seem to indicate that the results of this paper are of very limited practical significance. Of 1000 problems with 100 points each generated according to the uniform distribution in the unit square, only two had a realizable 2-factor and both of them failed to be a tour. Of 1000 problems with 50 points each, 87 had a realizable 2-factor but none of them was a tour. Of 1000 problems with 30 points each, 283 had a realizable 2-factor and three had a necklace-tour. Surprisingly, the probability that a point set uniformly distributed in the unit square has a realizable 1-factor (that is, a perfect matching) seems to be less than the likelihood of a realizable 2-factor (but greater than the likelihood of a realizable tour): of 1000 problems with 100 points each, none had a realizable 1-factor, of 1000 problems with 50 points each, 11 had a realizable 1-factor, and of 1000 problems with 30 points each, 78 had a realizable 1-factor.

We believe that the linear programming result of Section 5.1 is of independent interest. It is not hard to see, that the ideas generalize to arbitrary systems with m inequalities and n variables, if there are at most two variables in each inequality and the variables occur only with coefficients +1 or −1. By the presented technique, such a system can be solved in $O(n \min\{m, n^2\} + m)$ time and $O(m)$ space. Note, that in this restricted situation we can always detect all but at most $O(n^2)$ inequalities to be redundant.

It is interesting that the test of Section 5.1, which decides whether or not a given tour is a necklace-tour, could be performed in asymptotically less than $O(n^2)$ time if certain properties of the family of graphs $\mathcal{G}^{(2)}$ for point sets in the plane were known. For example, if these graphs have $n^{1/2}$-separators, then a test can be performed in $O(n^{3/2})$ time, which compares favorably with the $O(n^2)$ time complexity proved in this paper. This is because the single source shortest path problem can then be solved within this complexity bound (see [18]). The existence of such separators has recently been established by Rote [26].

References

[1] D. Avis and J. Horton, Remarks on the sphere of influence graph, in: J.E. Goodman et al., eds., *Proc. Conf. on Discrete Geometry and Convexity*, New York, *Ann. New York Acad. Sci.* **440** (1985) 323–327.

[2] E. Balas, Integer and fractional matchings, *Ann. Discrete Math.* **11** (1981) 1–13.

[3] P. Bateman and P. Erdös, Geometrical extrema suggested by a lemma of Besicovitch, *Amer. Math. Monthly* **58** (1951) 306–314.

[4] J.L. Bentley and T.A. Ottmann, Algorithms for reporting and counting geometric intersections, *IEEE Trans. Comput.* **C-28** (1979) 643–647.

[5] C. Berge, *Graphes et Hypergraphes* (Dunod, Paris, 2nd ed., 1973).

[6] U. Derigs and A. Metz, On the use of optimal fractional matchings for solving the (integer) matching problem, *Computing* **36** (1986) 263–270.

[7] E.W. Dijkstra, A note on two problems in connexion with graphs, *Numer. Math.* **1** (1959) 269–271.

[8] R.T. Duffin, On Fourier's analysis of linear inequality systems, *Math. Programming Stud.* **1** (1974) 71–95.

[9] H. Edelsbrunner, L.G. Guibas and J. Stolfi, Optimal point location in a monotone subdivision, *SIAM J. Comput.* **15** (1986) 317–340.

[10] J. Edmonds and R.M. Karp, Theoretical improvements in algorithmic efficiency for network flow problems, *J. Assoc. Comput. Mach.* **19** (1972) 248–264.

[11] J. Fourier, *Mémoires de l'Académie des Sciences* **7** (1827) 47–60.

[12] M.L. Fredman and R.E. Tarjan, Fibonacci heaps and their uses in improved network optimization algorithms, *J. Assoc. Comput. Mach.* **34** (1987) 596–615; also in: *Proc. 25th Ann. IEEE Symp. on Foundations Computer Science* (1984) 338–346.

[13] H.N. Gabow, An efficient reduction technique for degree-constrained subgraph and bidirected network flow problems, in: *Proc. 15th Ann. ACM Symp. on Theory Computing* (1983) 448–456.

[14] M.R. Garey and D.S. Johnson, *Computers and Intractability—A Guide to the Theory of NP-Completeness* (Freeman, San Francisco, 1979).

[15] V. Kantabutra, Traveling salesman cycles are not always subgraphs of Voronoi duals, *Inform. Process. Lett.* **16** (1983) 11–12.

[16] E.L. Lawler, *Combinatorial Optimization: Networks and Matroids* (Holt, Rinehart, and Winston, New York, 1976).

[17] D.T. Lee, On *k*-nearest neighbor Voronoi diagrams in the plane, *IEEE Trans. Comput.* **C-31** (1982) 478–487.

[18] R.J. Lipton, D.J. Rose and R.E. Tarjan, Generalized nested dissection, *SIAM J. Numer. Anal.* **16** (1979) 346–358.

[19] N. Megiddo, Towards a genuinely polynomial algorithm for linear programming, *SIAM J. Comput.* **12** (1983) 347–353.

[20] K. Mehlhorn, *Graph Algorithms and NP-Completeness* (Springer, Heidelberg, 1984).

[21] J.B. Orlin, Genuinely polynomial simplex and non-simplex algorithms for the minimum cost flow problem, Working paper No. 1615-84, Sloan School of Management, MIT, MA, 1984.

[22] C.H. Papadimitriou, The Euclidean TSP is NP-complete, *Theoret. Comput. Sci.* **4** (1977) 237–244.

[23] F.P. Preparata and M.I. Shamos, *Computational Geometry—An Introduction* (Springer, New York, 1985).

[24] E.R. Reifenberg, A problem on circles, *Math. Gaz.* **32** (1948) 290–292.

[25] A.G. Reinhold, Some results on minimal covertex polygons, Manuscript written under the supervision of Fred Supnick, City College of New York, 1965.

[26] G. Rote, Two solvable cases of the traveling salesman problem, Dissertation, Technische Universität Graz, Institut für Mathematik, 1988.

[27] D. Sanders, On extreme circuits, Ph.D. Thesis, City College of New York, 1968.

[28] M.I. Shamos, Computational geometry, Ph.D. Thesis, Dept. Computer Science, Yale Univ., New Haven, CO, 1978.

[29] R. Shostak, Deciding linear inequalities by computing loop residues, *J. Assoc. Comput. Mach.* **28** (1981) 769–779.

[30] F. Supnick, A class of combinatorial extrema, *Ann. New York Acad. Sci.* **175** (1970) 370–382.

[31] N. Tomizawa, On some techniques useful for solution of transportation network problems, *Networks* **1** (1972) 179–194.