Worst-Case Greedy Matchings in the Unit *d*-Cube

Timothy Law Snyder
Department of Computer Science, Georgetown University, Washington, DC 20057

J. Michael Steele
Program in Statistics and Operations Research, Princeton University, Princeton, New Jersey 08540

The worst-case behavior of greedy matchings of n points in the unit d-cube, where $d \geq 2$, is analyzed. The weighting function is taken to be the α'th power of Euclidean distance, where $0 < \alpha < d$. It is proved that the asymptotic growth rate of the weight of such a greedy matching is exactly $\beta n^{(d-\alpha)/d}$, where β is a positive constant that depends on the parameters α and d. Included in the analysis is a minimax theorem equating the worst-case behaviors of matchings resulting from greedy algorithms that, when ordering edges for the greedy process, break ties in different ways.

1. INTRODUCTION

The purpose of this paper is to determine the asymptotic behavior of the total weight of a worst-case Euclidean matching obtained by the greedy algorithm. Before reviewing the results that motivate this investigation, we will first make our problem precise and state the details of our main results.

A *matching* of the graph $G = (V, E)$ is a subset M of the edges of G such that a vertex of V is incident with at most one edge of M. Since all our matchings will be subsets of the complete graph on $V_n = \{x_1, x_2, \ldots, x_n\}$, a set of n points in the unit d-cube, we will refer to a matching as a *matching of the point set* V_n.

A simple heuristic for computing an approximate minimal matching of a given set of points is the *greedy algorithm*, which iteratively forms a matching M of the point set V_n by initializing M as the empty set and considering the edges of the complete graph on V_n in order of nondecreasing weight. At each iteration, a candidate edge e is placed into M if neither endpoint of e is incident with an edge of M. A matching formed using the greedy heuristic is called a *greedy matching*. In cases where there exist more than one choice of orderings of candidate edges in nondecreasing order, we will often find it beneficial to

NETWORKS, Vol. 20 (1990) 779–800

CCC 0028-3045/90/060779-022$04.00

consider the specific ordering of candidate edges to be independent of the greedy process, in which case we will refer to applying the greedy algorithm to a given candidate edge sequence. Although the greedy heuristic can also be applied to form an approximate maximum-weight matching in an obvious way, we shall always use the term greedy in the context of minimal matchings.

Let $e = \{x_i, x_j\}$ denote an edge of the complete graph on V_n. We will use $|e|$ to denote the usual Euclidean length $|x_i - x_j|$, and we will be concerned with the edge weighting function w defined by $w(e) = |e|^\alpha$, where $0 < \alpha < d$. *Power-weighted* edges are edges whose length is weighted according to the weighting function w. Our study of power-weighted edges is motivated by simulations in which squared Euclidean edge weights are used in lieu of Euclidean lengths. Such weights are often used to avoid the extra computational overhead of square-root calculations. Although the case $\alpha = 1$ is still of primary concern, the study of the more general case $0 < \alpha < d$ has independent interest, and, in addition, this increased generality helps to illustrate some key features of our method.

In particular, one should note that the edge weighting function w typically fails to provide a metric on V_n since the triangle inequality can fail to hold for all $\alpha \neq 1$. The features of w that turn out to be essential are that w is a *homogeneous*, *monotone* function of Euclidean length. Because of the key role played by rescaling and approximate self-similarity, homogeneity is critical.

Our fundamental object of interest is

$$\rho_g(n) = \sup_{\substack{V_n \subset [0,1]^d \\ |V_n| = n}} \max_G \left\{ \sum_{e \in G} |e|^\alpha : G \text{ is a greedy matching of } V_n \right\}. \tag{1.1}$$

We think of $\rho_g(n)$ as the maximum possible total weight of a greedy matching of n points in the unit d-cube. For $\epsilon > 0$, we will call a greedy matching having total weight at least $\rho_g(n) - \epsilon$ an *ϵ-worst case* greedy matching with power-weighted edges. We will subsequently refer to an ϵ-worst-case greedy matching simply as a *worst-case* greedy matching, and the set of points producing such a matching will be called a *worst-case point configuration*.

The internal maximum in the definition of $\rho_g(n)$ is a consequence of the fact that if there exist edges of the complete graph on V_n that are of identical length, then there can exist more than one greedy matching of V_n.

Our main asymptotic result for ρ_g is summarized in the following theorem.

Theorem 1. *There exists a positive constant β_g depending on the dimension $d \geq 2$ and the edge-weighting function $w(e) = |e|^\alpha$, where $0 < \alpha < d$, such that as $n \to \infty$,*

$$\rho_g(n) \sim \beta_g n^{(d-\alpha)/d}. \tag{1.2}$$

This result provides the first determination of the exact asymptotic rate of growth of the function ρ_g, and it also represents the first investigation of greedy matchings under power-weighted edges.

Some analogous results for the minimal spanning tree and for the optimal traveling salesman tour of n points in the d-cube are given in Steele and Snyder [12], but the lack of minimality of a greedy matching and the fact that there can exist more than one greedy matching of a given set of points force one to tackle a number of new issues in the analysis of the greedy heuristic.

An issue that must be addressed is that of tied edges, or edges of identical Euclidean length in the complete graph on V_n. Different lists of the edges of the complete graph on V_n arranged in order of nondecreasing Euclidean length can produce greedy matchings of significantly different weights, so we obtain a family of greedy matchings of V_n that depend on the resolution (or ordering) of the ties.

The definition of ρ_g in (1.1) considers the *worst-case weight* of a greedy matching formed by an algorithm that breaks ties in the worst possible way, forming a maximum-weight greedy matching of V_n. We can also define the worst-case performance of an *optimal* greedy algorithm in the sense that the algorithm resolves ties so as to attain the cheapest-possible greedy matching:

$$\hat{\rho}_g(n) = \sup_{\substack{V_n \subset [0,1]^d \\ |V_n| = n}} \min_G \left\{ \sum_{e \in G} |e|^\alpha : G \text{ is a greedy matching of } V_n \right\}. \tag{1.3}$$

Fortunately, the resolution of ties turns out to be of no consequence for worst-case point configurations; this is indicated by the following result.

Theorem 2. *For all $n \geq 1$, one has that*

$$\hat{\rho}_g(n) = \rho_g(n). \tag{1.4}$$

All told, it may be somewhat surprising that the two quantities of Theorem 2 are equal. In addition to telling us that one has more than the asymptotic equivalence $\hat{\rho}_g(n) \sim \rho_g(n)$ as $n \to \infty$, the result tells us that the worst-case weighted matchings of all greedy algorithms are equivalent for all $n \geq 1$.

Previous work on greedy matchings has focused on bounding the quantity $\rho_g(n)$. For example, Avis [1] used a hexagonal lattice sphere packing to show that in $d = 2$ we have that $\rho_g(n) \geq 0.8474\sqrt{n}$, and by developing upper bounds on the minimum distances between points of the unit square, Avis [1] showed that $\rho_g(n) \leq 1.074\sqrt{n} + O(\log n)$. It is essential to note, however, that these big-O and Ω bounds on $\rho_g(n)$ do not address the issue of convergence of $\rho_g(n)/n^{(d-\alpha)/d}$ to a constant.

The only known exact asymptotic results for greedy matchings is the probabilistic result of Avis et al. [3]. If we let G_n be a greedy matching of $\{X_1, X_2, \ldots, X_n\} \subset [0, 1]^d$, where the X_i are independent, identically distributed random vectors and where $d \geq 2$, then Avis et al. established the following:

Theorem. *With probability one as $n \to \infty$, one has*

$$G_n \sim c(d) n^{(d-1)/d} \int_{\mathbb{R}^d} [f(x)]^{(d-1)/d} \, dx. \tag{1.5}$$

Here, $c(d)$ is a constant depending on the dimension d and f is the density of the absolutely continuous part of the distribution of the X_i, where $1 \le i \le n$. We note that the rate of growth of G_n in this theorem is similar in form to the rate of growth of ρ_g in our Theorem 1, and, indeed, the existence of this probabilistic result motivates our investigation of the corresponding worst-case problem.

The survey of matching heuristics given by Avis [2] goes further into the history and motivation of our problem than we can here, but we should mention some salient aspects of earlier work. Euclidean greedy matchings have been used in applications such as plotting a graph using a mechanical pen plotter [2,8,10] and as a heuristic in Christofides' algorithm for finding an approximate traveling salesman tour of n points whose edges satisfy the triangle inequality [4]. For graphs with large numbers of vertices such as those from applications in VLSI, the well-known $O(n^4)$ minimal matching algorithm of Edmonds [5] and even the $O(n^3)$ refinement of Gabow [6] and Lawler [9] are too slow for large numbers of points, making faster heuristics desirable.

In what follows, we will at all times focus on *worst-case* results for points in $[0, 1]^d$, without any probabilistic assumptions. In the next section, we spell out the precise properties that are needed by ρ_g in order to establish the asymptotics reflected in Theorem 1. The heart of the matter is to show that ρ_g satisfies the approximate recursion relation (2.1(b)) of Lemma 2.1. The first step of this verification is the construction of a point set for which a greedy matching nearly obtains the worst-case weight; this is accomplished in Section 3, and it leads to a preliminary recursion that works together with the geometric inequalities of Sections 4 and 5 to verify the hypotheses of Lemma 2.1. Section 4 bounds the maximum number of relatively long edges of a greedy matching in the unit d-cube, and Section 5 bounds some incremental rates of growth of ρ_g. These ingredients are brought together in Section 6 to prove Theorem 1.

The proof of Theorem 2 is handled in Sections 7–9, and Section 10 offers concluding remarks, including some comments on open problems.

2. TARGET RECURSIONS

Before attacking the analysis of ρ_g, we need a couple of general lemmas that are not directly related to matchings. Although the first lemma looks technical, it tells us just which properties of ρ_g must be established. The first such *target recursion*, relation (2.1(a)), expresses a slow incremental rate of growth condition, whereas the second, (2.1(b)), guides the majority of the subsequent work.

Lemma 2.1. *If $\rho(1) = 0$, $0 < \alpha < d$, and there exists some $c \ge 0$ such that for all $m \ge 1$ and $k \ge 1$:*

$$\begin{aligned} &\text{(a)}\ \ \rho(n+1) \le \rho(n) + cn^{-\alpha/d}\ \textit{and} \\ &\text{(b)}\ \ m^{d-\alpha}\rho(k) - m^{d-\alpha}k^{(d-\alpha)/d}r_\alpha(k) \le \rho(m^d k), \end{aligned} \tag{2.1}$$

where $r_\alpha(k) \to 0$ *as* $k \to \infty$, *then as* $n \to \infty$,

$$\rho(n) \sim \beta n^{(d-\alpha)/d}$$

for a constant $\beta \ge 0$.

Proof. From the hypothesis (2.1a), we first note that for $1 \le i < j < \infty$ we have

$$\begin{aligned} \rho(j) - \rho(i) &= \sum_{k=i}^{j-1} \{\rho(k+1) - \rho(k)\} \\ &\le c \sum_{k=i}^{j-1} k^{-\alpha/d} \\ &\le c \int_{i-1}^{j-1} x^{-\alpha/d}\,dx \\ &\le c' \frac{d}{d-\alpha}\{j^{(d-\alpha)/d} - i^{(d-\alpha)/d}\}, \end{aligned} \tag{2.2}$$

where the constant c' depends on α. Letting $i=1$ and $j=n$ in Inequality (2.2) and using the fact that $\rho(1)=0$ shows us that $\rho(n)$ satisfies $\rho(n) \le c'(d/(d-\alpha))n^{(d-\alpha)/d}$, so if we define

$$\Psi(k) = \frac{\rho(k)}{k^{(d-\alpha)/d}}, \tag{2.3}$$

we see that $\Psi(k)$ is bounded. This allows us to set

$$\gamma = \limsup_{k \to \infty} \Psi(k) < \infty. \tag{2.4}$$

Dividing Inequality (2.1(b)) by $m^{d-\alpha}k^{(d-\alpha)/d}$ gives us the simplified relation

$$\Psi(k) - r_\alpha(k) \le \Psi(m^d k) \tag{2.5}$$

for all $m \ge 1$ and $k \ge 1$. Now, given any positive real number ϵ, we can choose a k_ϵ such that $\gamma - \epsilon \le \Psi(k_\epsilon)$ and $r_\alpha(k_\epsilon) \le \epsilon$, from which (2.5) becomes

$$\gamma - 2\epsilon \le \Psi(m^d k_\epsilon) \tag{2.6}$$

for all $m \ge 1$.

We now consider the intervals $j_m \le n \le j_{m+1}$, where $j_m = m^d k_\epsilon$. To bound the absolute difference $|\Psi(n) - \Psi(j_m)|$, we first use Inequality (2.2) to check how much $\rho(n)$ can differ from $\rho(j_m)$:

$$\sup_{j_m \le n \le j_{m+1}} |\rho(n) - \rho(j_m)| = |\rho(j_{m+1}) - \rho(j_m)|$$

$$\le c' \frac{d}{d-\alpha} k_\epsilon^{(d-\alpha)/d}\{(m+1)^{d-\alpha} - m^{d-\alpha}\}. \quad (2.7)$$

Using the mean value theorem, we then bound $(m+1)^{d-\alpha} - m^{d-\alpha}$, obtaining two cases that depend on the size of the parameter α, namely,

$$(m+1)^{d-\alpha} - m^{d-\alpha} \le \begin{cases} (d-\alpha)(m+1)^{d-1-\alpha}, & \text{if } \alpha \le d-1; \\ (d-\alpha)m^{d-1-\alpha}, & \text{if } \alpha > d-1. \end{cases} \quad (2.8)$$

We can then express the bound (2.7) in terms of Ψ to get:

$$|\Psi(n) - \Psi(j_m)| \le \frac{|\rho(n) - \rho(j_m)|}{j_m^{(d-\alpha)/d}}$$

$$\le \begin{cases} c'd\dfrac{1}{m}\left(\dfrac{m+1}{m}\right)^{d-\alpha}, & \text{if } \alpha \le d-1; \\ c'd\dfrac{1}{m}, & \text{if } \alpha > d-1 \end{cases} \quad (2.9)$$

$$\le 2^d c'd\frac{1}{m}$$

for $j_m \le n \le j_{m+1}$, where we have bounded $(m+1)^{d-\alpha}/m^{d-\alpha}$ by $2^{d-\alpha}$. Hence,

$$-2^d c'd\frac{1}{m} \le \Psi(n) - \Psi(j_m) \quad (2.10)$$

for $j_m \le n \le j_{m+1}$.

Returning to (2.6), we now see that

$$\gamma - 2^d c'd\frac{1}{m} - 2\epsilon \le \Psi(n) \quad (2.11)$$

for any $j_m \le n \le j_{m+1}$. Thus $\liminf_{k\to\infty} \Psi(k) \ge \gamma - 2\epsilon$, and since $\epsilon > 0$ is arbitrary, we can conclude that

$$\gamma \le \liminf_{n\to\infty} \Psi(n), \quad (2.12)$$

which implies that $\Psi(n) \to \gamma$ as $n \to \infty$ and completes the lemma with $\beta = \gamma$. ■

Our next lemma tells us that from any n points in $[0, 1]^d$ we can always find a pair of points that are relatively close together. Even though this is a well-known fact, we present a brief proof here for completeness.

Lemma 2.2. *There is a constant c_d such that for any $\{x_1, x_2, \ldots, x_n\} \subset [0, 1]^d$, where $n \geq 2$, one has*

$$|x_i - x_j| \leq c_d n^{-1/d} \tag{2.13}$$

for some x_i and x_j, where $1 \leq i < j \leq n$.

Proof. We proceed using a simple packing argument. Cover each point x_i with a d-ball of radius r and center x_i. Such a ball has a volume of $\omega_d r^d$, where ω_d is the volume of the unit d-ball. Suppose that all these balls were nonintersecting. Then, each ball would cover at least $2^{-d}\omega_d r^d$ of $[0, 1]^d$ (even if the ball was centered at a corner of the hypercube). The n balls would cover at least $2^{-d}\omega_d r^d n$. Since $[0, 1]^d$ has unit volume, we have $2^{-d}\omega_d r^d n \leq 1$. Hence, $|x_i - x_j| \leq 4\omega_d^{-1/d} n^{-1/d}$ for some x_i and x_j, where $1 \leq i < j \leq n$. This proves the lemma with $c_d = 4\omega_d^{-1/d}$. ∎

One can improve the constant of this lemma, but the simply derived constant c_d is sufficient for our purposes.

In the next section we partition the d-cube to construct a point set that attains a nearly worst-case greedy matching. This will give us an initial recursion that is a first stab at the second target recursion of Lemma 2.1.

3. CONSTRUCTION OF A NEARLY WORST-CASE GREEDY MATCHING

The asymptotic analysis of ρ_g will be obtained by giving a recursive construction that demonstrates that ρ_g satisfies the second target recursion of Lemma 2.1. Although the construction itself is easy to lay out, the verification that the construction suffices will require a rather detailed understanding of the combinatorial geometry associated with Euclidean greedy matchings.

We begin the recursive construction by dividing the d-cube $Q = [0, 1]^d$ into m^d equally sized *cells* Q_i, where $1 \leq i \leq m^d$ and each cell has side length $1/m$. The union of the boundaries of the Q_i will be denoted by H; in other words, $H = \bigcup_{i=1}^{m^d} \partial Q_i$. Also, for δ satisfying $0 < \delta < 1/m$, let H^δ denote the set of points of $[0, 1]^d$ that are within $\delta/2$ of H; in other words, H^δ is the *grating* H fattened to a width of δ. From this, we can define a *subcell* Q_i^δ of each Q_i by $Q_i^\delta = Q_i - H^\delta$.

Now, by the definition of $\rho_g(k)$, for all $\epsilon > 0$ there is a set of k points $S = \{x_1, x_2, \ldots, x_k\}$ in $[0, 1]^d$ for which the greedy algorithm yields a matching of total weight at least $\rho_g(k) - \epsilon$. By scaling and translation, we can therefore produce for each $1 \leq i \leq m^d$ a set $S_i \subset Q_i^\delta$ of k points such that the greedy algorithm applied to S_i can produce a matching of weight at least $(m^{-1} - \delta)(\rho_g(k) - \epsilon)$, since each side of Q_i^δ has width $(m^{-1} - \delta)$. To do so, the greedy algorithm must break ties to form an ordering that yields a maximum-weight greedy matching on the point set S_i. Let M_i denote this greedy matching. Next, we will construct a specific greedy matching M of the $m^d k$ points $\cup_{i=1}^{m^d} S_i$ that is closely related to the matchings M_i of the individual S_i. The proof that ρ_g satisfies the second target recursion of Lemma 2.1 will come from building a relationship between the total weight of M and the total weight of the collection of the M_i, where $1 \leq i \leq m^d$.

We form M by applying the greedy heuristic to a special ordering σ of the edges of the complete graph K on the $m^d k$ points in $[0,1]^d$. To begin the definition of σ, we first place the edges of K in order of nondecreasing Euclidean length, and we resolve any ties that may occur by the following sequential process.

The first step in the tie resolution process focuses on the edges that lie entirely within subcells; in other words, all edges e such that e is contained entirely in the union $\cup_{i=1}^{m^d} Q_i^\delta$. For any set of edges of K that have the same length λ, i.e., for

$$E_\lambda = \{e \in K : |e| = \lambda, e \subset Q_i^\delta, \text{ for some } 1 \leq i \leq m^d\},$$

the edges of E_λ are first ordered according to the index of the subcells in which they are contained. Thus, the edges of Q_1^δ appears first, the edges of Q_2^δ appear second, and so on, concluding with the edges of $Q_{m^d}^\delta$.

The ordering σ of the edges of K will be completely determined once we specify an ordering of $E_\lambda \cap \{e : e \subset Q_i^\delta\}$ for each $1 \leq i \leq m^d$ and for each λ such that $\lambda = \{|e| : e \text{ is an edge of } K\}$. To do this, we recall that each matching M_i of $S_i \subset Q_i^\delta$ is precisely the greedy matching corresponding to a given ordering σ_i of $\{e : e \subset Q_i^\delta\}$. Since elements of $E_\lambda \cap \{e : e \subset Q_i^\delta\}$ can appear in the same order in σ as they do in σ_i without violating the nondecreasing ordering of σ, we choose for each i and each λ the ordering of $E_\lambda \cap \{e : e \subset Q_i^\delta\}$ specified by σ_i to complete our specification of σ.

The main consequence of this construction is that if e and e' both are edges of the complete graph K on $\cup_{i=1}^{m^d} S_i$, and if $e \subset Q_i^\delta$ and $e' \subset Q_i^\delta$ for some $1 \leq i \leq m^d$, then e precedes e' in the ordering σ if and only if e precedes e' in the ordering σ_i. In addition, if $e \subset Q_i^\delta$ and $e' \subset Q_j^\delta$ for $i \neq j$, then if $|e| < |e'|$, we have e preceding e' in the ordering σ. Finally, if $e \subset Q_i^\delta$ and $e' \subset Q_j^\delta$ for $i \neq j$ and if $|e| = |e'|$, then e precedes e' if and only if $i < j$.

We now consider the process of simultaneously forming the matchings M_i and M using the orderings σ_i and σ, respectively, where $1 \leq i \leq m^d$. Consider first a smallest edge e of σ, and note that if $|e| < \delta$, then e must belong to M and also to some M_i. Proceeding successively, we note that any edge e that satisfies $|e| < \delta$ and is chosen for some M_i must also be chosen for M, by the construction of the ordering σ. Consequently, we arrive at the basic inclusion

$$\cup_{i=1}^{m^d} M_i \subseteq M \cup \{e : e \in \cup_{i=1}^{m^d} M_i \text{ and } |e| \geq \delta\}, \tag{3.1}$$

which tells us that the associated edge weights must satisfy

$$m^d(m^{-1}-\delta)^\alpha(\rho_g(k)-\epsilon)\le\sum_{e\in M}|e|^\alpha+\sum_{e\in F}|e|^\alpha\le\rho_g(m^dk)+\sum_{e\in F}|e|^\alpha, \tag{3.2}$$

where $F=\cup_{i=1}^{m^d}\{e: e\in M_i \text{ and } |e|\ge\delta\}$. The second inequality in (3.2) comes from the fact that $\rho_g(m^dk)$ is an upper bound on any greedy matching of m^dk points in $[0,1]^d$.

To turn the key recursion (3.2) into the inequality that can serve to justify the second target recursion of Lemma 2.1, we need to find an upper bound on the total weight that a greedy matching can assign to its long edges, such as the members of F. This is the task of the next section.

4. A BOUND ON LONG GREEDY EDGES

In this section, lemmas leading to a bound on the number of relatively long edges of a greedy matching in $[0,1]^d$ are developed. The first lemma shows that there is an *a priori* bound on the total weight of any subset of k edges of a greedy matching.

It will be convenient henceforth to use $L_\alpha(E)$ to denote the sum of the weights $|e|^\alpha$ for all $e\in E$, where E is any subset of the edges of the complete graph on V_n.

Lemma 4.1. *There is a constant c_3 such that any subset E of the edges of a greedy matching of $\{x_1,x_2,\ldots,x_n\}\subset[0,1]^d$ satisfies*

$$L_\alpha(E)\le c_3|E|^{(d-\alpha)/d}. \tag{4.1}$$

Proof. Let $s=\lfloor n/2\rfloor$. We label the edges $e_1,e_2,\ldots,e_s$ of a greedy matching M by the order in which they are chosen by the greedy algorithm that forms the matching. If we let $k=|E|$ and $e_{i_1},e_{i_2},\ldots,e_{i_k}$ be the edges of E, then by the relabeling of the e_i we have that

$$L_\alpha(E)=\sum_{j=1}^{k}|e_{i_j}|^\alpha\le\sum_{i=s-k}^{s-1}|e_{i+1}|^\alpha. \tag{4.2}$$

For $0\le i\le s-1$, before e_{i+1} is chosen by the greedy algorithm for inclusion into M, there are exactly $n-2i$ points of $\{x_1,x_2,\ldots,x_n\}$ yet to be matched. By Lemma 2.2, we thus have

$$\begin{aligned}\sum_{i=s-k}^{s-1}|e_{i+1}|^\alpha&\le c_2^\alpha\sum_{i=s-k}^{s-1}(n-2i)^{-\alpha/d}\\&\le c_2^\alpha\sum_{i=1}^{k}(2i)^{-\alpha/d}\\&\le c_2^\alpha 2^{-\alpha/d}\left(1+\int_1^k x^{-\alpha/d}\,dx\right)\\&\le c_3k^{(d-\alpha)/d},\end{aligned}\tag{4.3}$$

where $c_3 = d/(d-\alpha)\, 2^{-\alpha/d} c_2^{\alpha}$. ■

Although we will not make use of the following fact, one should note that Lemma 4.1 cannot be significantly improved. In particular, if n points of $[0, 1]^d$ are laid out in a regular hexagonal lattice, then any set of k edges of *any* matching will have weight on the order of $kn^{-\alpha/d}$, which is $\Omega(n^{(d-\alpha)/d})$ when k is of order n.

Our next lemma bounds the number of edges of Euclidean length x or greater in a greedy matching of n points of the cube. We first define $\nu_g(x, n)$ to be the largest j such that there exists some point set $V_n = \{x_1, x_2, \dots, x_n\}$ having a greedy matching with j edges greater than or equal to x in Euclidean length, i.e.,

$$\nu_g(x, n) = \max_{\substack{V_n \subset [0,1]^d \\ |V_n| = n}} \max_{M} |\{e \in M : |e| \geq x \text{ and } M \text{ is a greedy matching of } V_n\}|. \tag{4.4}$$

The internal maximum, of course, is necessitated by the possibility of more than one greedy matching of a given point set V_n.

An intriguing fact is that $\nu_g(x, n)$ can be bound independent of n, the number of points.

Lemma 4.2. *There is a constant c_4 such that, for all $x > 0$,*

$$\nu_g(x, n) \leq c_4 x^{-d}. \tag{4.5}$$

Proof. Let M be a greedy matching of $\{x_1, x_2, \dots, x_n\}$, and set $\phi_M(x) = |\{e \in M : |e| \geq x\}|$. Label the edges of M having length x or greater in the order of their selection by the greedy algorithm as $e_1, e_2, \dots, e_{\phi_M(x)}$, i.e., $|e_1| \leq |e_2| \leq \cdots \leq |e_{\phi_M(x)}|$.

By the nature of the greedy algorithm, for any $1 \leq i \leq \phi_M(x)$, we can choose an endpoint p of e_i so that a sphere of radius x centered at p contains no members of the set of endpoints of the e_j, where $i + 1 \leq j \leq \phi_M(x)$. Therefore, if we consider spheres of radius $x/2$ for all e_i, where $1 \leq i \leq \phi_M(x)$, then none of the spheres can intersect.

If ω_d is the volume of the unit sphere in $[0, 1]^d$, then we have $\phi_M(x)$ spheres, each of volume $\omega_d(x/2)^d$. Consider the sphere associated with the edge e_i, and let A_i be the portion of sphere that intersects with $[0, 1]^d$. If $\mu(A_i)$ is the volume of A_i, then $\sum_{i=1}^{\phi_M(x)} \mu(A_i) \geq \phi_M(x)\omega_d 2^{-d}(x/2)^d$, where the quantity on the right assumes that each A_i is centered at the corner of the d-cube. This yields $\phi_M(x)\omega_d 2^{-d}(x/2)^d \leq 1$, which proves the lemma with $c_4 = 4^d/\omega_d$. ■

Using this bound on the maximum number of long edges along with the bound of the previous lemma on the total weight of any subset of k edges in a greedy matching, we are now in a position to exploit the relationship between M and the M_i given by Inequality (3.2). To bound the total weight of the edges in the set $F = \cup_{i=1}^{m^d} \{e : e \in M_i \text{ and } |e| \geq \delta\}$, we note that M_i is a matching of k points in a cube of side length $m^{-1} - \delta$. By scaling, we see from (4.1) that

$$\sum_{\substack{e \in M_i \\ |e| \geq \delta}} |e|^\alpha \leq c_3(m^{-1} - \delta)^\alpha |\{e : e \in M_i \text{ and } |e| \geq \delta\}|^{(d-\alpha)/d}. \tag{4.6}$$

Now, if we let $\tilde{\nu}_g(x, n)$ be the maximum number of edges of length at least x in a greedy matching of n points in $[0, t]^d$, then we have the identity

$$\tilde{\nu}_g(tx, n) = \nu_g(x, n). \tag{4.7}$$

Letting $t = m^{-1} - \delta$ and $x = \delta/t$, we see from (4.7) and the bound of Lemma 4.2 that

$$|\{e : e \in M_i \text{ and } |e| \geq \delta\}| \leq \tilde{\nu}_g(\delta, n) \leq c_4 \delta^{-d}(m^{-1} - \delta)^d, \tag{4.8}$$

so (4.6) becomes

$$\sum_{\substack{e \in M_i \\ |e| \geq \delta}} |e|^\alpha \leq c_3 c_4^{(d-\alpha)/d}(m^{-1} - \delta)^{d-\alpha} \delta^{\alpha-d}. \tag{4.9}$$

Using (4.9) to bound $\Sigma_{e \in F} |e|^\alpha$, we can now simplify our key recursion (3.2) to obtain

$$m^d(m^{-1} - \delta)^\alpha(\rho_g(k) - \epsilon) - c_3 c_4^{(d-\alpha)/d} m^d (m^{-1} - \delta)^d \delta^{\alpha-d} \leq \rho_g(m^d k), \tag{4.10}$$

and, since δ and ϵ are positive, we conclude that

$$m^d(m^{-1} - \delta)^\alpha \rho_g(k) - c_3 c_4^{(d-\alpha)/d} \delta^{\alpha-d} \leq \rho_g(m^d k). \tag{4.11}$$

In Section 6 we will show that δ can be chosen so that the relation (4.11) satisfies our second target recursion. The next section deals with the incremental rate of growth of ρ_g in order to satisfy our first target recursion.

5. LEMMAS ON GREEDY INCREMENTAL GROWTH

The first of the two target recursions required by Lemma 2.1 is a bound on the incremental rate of growth of the function ρ_g. The next lemma establishes an incremental rate of growth "by two's," while allowing for a nonmonotonicity of ρ_g.

Lemma 5.1. *There is a constant c_5 such that for all $0 < \alpha < d$ one has*

$$\rho_g(n + 1) \leq \rho_g(n - 1) + c_5 n^{-\alpha/d}, \textit{ for all } n \geq 1; \tag{5.1}$$

and

$$\rho_g(n+1) \leq \rho_g(n), \textit{ for all } n \geq 2, \textit{ where } n \textit{ is even.} \tag{5.2}$$

Proof. Let S be an arrangement of $n+1$ points in $[0,1]^d$ such that a greedy matching M_{n+1} of S has weight $L_\alpha(M_{n+1}) \geq \rho_g(n+1) - \epsilon$, where $\epsilon > 0$. Let $\{x_i, x_j\}$ be a shortest edge of the complete graph on S. By Lemma 2.2 and the nature of the greedy heuristic, $\{x_i, x_j\} \in M_{n+1}$ and $|x_i - x_j|^\alpha \leq c_2^\alpha (n+1)^{-\alpha/d} \leq c_2^\alpha n^{-\alpha/d}$.

To prove Inequality (5.1), delete both x_i and x_j and obtain the matching M, which is M_{n+1} with the edge $\{x_i, x_j\}$ removed; we then have that

$$\rho_g(n+1) - \epsilon \leq L_\alpha(M) + |x_i - x_j|^\alpha \leq L_\alpha(M) + c_2^\alpha n^{-\alpha/d}. \tag{5.3}$$

In addition, since $\{x_i, x_j\}$ is an edge of minimum length, M is a greedy matching of $\{x_1, x_2, \ldots, x_{i-1}, x_{i+1}, \ldots, x_{j-1}, x_{j+1}, \ldots, x_n\}$; hence, we have that $L_\alpha(M) \leq \rho_g(n-1)$. Inequality (5.3) then becomes $\rho_g(n+1) - \epsilon \leq \rho_g(n-1) + c_2^\alpha n^{-\alpha/d}$. This is true for any $\epsilon > 0$, proving (5.1) with $c_5 = c_2^\alpha$.

To prove (5.2), we first suppose that n is even and again let S be an arrangement of $n+1$ points with a greedy matching M_{n+1} of weight $L_\alpha(M_{n+1}) \geq \rho_g(n+1) - \epsilon$, where $\epsilon > 0$. Since $n+1$ is odd, then remains exactly one exposed point x^* of S following the formation of M_{n+1}. We delete this exposed point and form a greedy matching M_n on the remaining points of S such that M_n satisfies $L_\alpha(M_n) = \max\{L_\alpha(M) : M \text{ is a greedy matching of } S - \{x^*\}\}$.

Now, since there exists an ordering σ of the edges of the complete graph on $S - \{x^*\}$ such that the greedy algorithm pplied to σ produces M_{n+1} (this can be accomplished by deleting the edges incident with x^* from the ordering used to form M_{n+1} on S), we see that $L_\alpha(M_{n-1}) \leq L_\alpha(M_n)$. Consequently, we have

$$\rho_g(n+1) - \epsilon \leq L_\alpha(M_{n+1}) \leq L_\alpha(M_n) \leq \rho_g(n). \tag{5.4}$$

Since (5.4) is true for all $\epsilon > 0$, the proof is complete. ■

We remark that the case distinctions of Lemma 5.1 are unavoidable since $\rho_g(n)$ is not a monotone sequence.

Because Lemma 5.1 only gives us an incremental rate of growth of the function ρ_g "by two's," we will later see that we can apply Lemma 2.1 only to greedy matchings with power-weighted edges of sets of points of even cardinality. This problem is resolved by the following lemma, which allows us to compare the weight of a greedy matching of a set of points with the weight of a greedy matching of the same set of points augmented by an additional point.

Lemma 5.2. *Let M_{n+1} and M_n be greedy matchings using the weighting function $w(e) = |e|^\alpha$ of $S_{n+1} = \{x_1, x_2, \ldots, x_{n+1}\}$ and $S_n = \{x_1, x_2, \ldots, x_n\}$, respectively, where $x_i \in [0,1]^d$ for $1 \leq i \leq n+1$. Then,*

$$|L_\alpha(M_{n+1}) - L_\alpha(M_n)| \leq d^{\alpha/2}. \tag{5.5}$$

Proof. Order and label the $s = \binom{n+1}{2}$ edges of the complete graph on S_{n+1} so that $|e_1| \leq |e_2| \leq \cdots \leq |e_s|$.

We can iteratively and simultaneously form both M_{n+1} and M_n by considering each edge e_i, where $i = 1, 2, \ldots, s$, for inclusion in both M_{n+1} and M_n. We begin by initializing M_{n+1} and M_n as empty sets. If both endpoints of e_i are exposed in M_{n+1}, then e_i is added to M_{n+i}; in addition, if both endpoints of e_i are exposed in M_n and neither endpoint is the point x_{n+1}, then e_i is added to M_n.

On completion of this algorithm, we claim that exactly one of the following properties is true for each e_i:

(1) $e_i \in M_{n+1}$ and $e_i \in M_n$;
(2) $e_i \notin M_{n+1}$ and $e_i \notin M_n$;

or

(3) e_i belongs to a *monotone alternating path* of M_n and M_{n+1} originating at x_{n+1}, where a monotone alternating path P of M_n and M_{n+1} is a path of length at least one whose edges are monotonically nondecreasing in length and alternate from membership in one matching to membership in the other as each edge of the path is traversed. The originating edge of P with x_{n+1} as an endpoint is an edge of minimum length in the path. (For a more formal definition, see Snyder [11].)

We prove the claim by induction on i. For $i = 1$, if x_{n+1} is not an endpoint of e_1, then Property (1) is trivially satisfied. If x_{n+1} is an endpoint of e_1, then e_1 is itself an alternating monotone path originating at x_{n+1} and Property (3) is true, so we assume the claim is true for all e_j, where $j < i$, and consider the edge e_i.

If neither (1) nor (2) is satisfied, then we have two possibilities. In the first case, suppose $e_i \in M_n$ and $e_i \notin M_{n+1}$. Edge e_i can fail to be in M_{n+1} only if one of its endpoints is the endpoint of some edge $e_j \in M_{n+1}$. By the nature of the greedy algorithm, $j < i$ and $|e_j| \le |e_i|$, and by the inductive hypothesis, e_j is an edge of a monotone alternating path of M_n and M_{n+1} originating at x_{n+1}. Hence, e_i satisfies Property (3).

The second case occurs when $e_i \notin M_n$ and $e_i \in M_{n+1}$. One way this can happen is when x_{n+1} is an endpoint of e_i, in which case (3) is trivially satisfied. The other way in which this can occur is when x_{n+1} is not an endpoint of e_i. In this case, (3) is satisfied using the same argument we used in case one, proving the claim.

To complete the proof, we note that the only edges contributing to $|L_\alpha \times (M_{n+1}) - L_\alpha(M_n)|$ are those satisfying Property (3). Since by definition the monotone alternating path P originating at x_{n+1} must contain at least one edge of M_{n+1}, there can be only one such path by the definition of a matching, and since P is monotone,

$$|L_\alpha(M_{n+1}) - L_\alpha(M_n)| = \left| \sum_{\substack{e \in P \\ e \in M_{n+1}}} |e|^\alpha - \sum_{\substack{e \in P \\ e \in M_n}} |e|^\alpha \right|$$

$$\le \max\{|e|^\alpha : e \in P\} \le d^{\alpha/2}, \tag{5.6}$$

completing the proof. ■

A lemma that is closely related to Lemma 5.2 with $\alpha = 1$ is given in [3], where it is used in a probabilistic context.

6. GREEDY ASYMPTOTICS

We now use the combinatorial and geometric lemmas thus far established to show that ρ_g satisfies the target recursions of Lemma 2.1. We first note that since $\delta < 1/m$, we have that $\delta^{\alpha-d} < m^{-\alpha}\delta^{-d}$. Inserting this bound into the key relationship (4.11) and expanding the quantity $(m^{-1} - \delta)^{\alpha}$ gives us

$$m^{d-\alpha}\rho_g(k) - m^{d+1-\alpha}\delta\alpha\rho_g(k) + O(m^{d+2-\alpha}\delta^2)\rho_g(k) - c_3 c_4^{d-\alpha} m^{-\alpha}\delta^{-d} \le \rho_g(m^d k). \tag{6.1}$$

We then make the choice

$$\delta = \frac{1}{m}\alpha^{-1/(d+1)}k^{(\alpha-d)/d(d+1)} \tag{6.2}$$

to show that there exists a constant c_6 such that

$$m^{d-\alpha}\rho_g(k) - c_6 m^{d-\alpha}\alpha^{d/(d+1)}k^{(d-\alpha)/(d+1)} \le \rho_g(m^d k) \tag{6.3}$$

for all m and k greater than or equal to one. Since $0 < \alpha < d$, we see that our choice of δ satisfies the requirement that $0 < \delta < 1/m$. Moreover, the recursion (6.3) satisfies the second target recursion of Lemma 2.1 with the remainder function $r_\alpha(k) = c_6\alpha^{d/(d+1)}k^{(\alpha-d)/d(d+1)}$, which goes to zero as $k \to \infty$ since α is fixed and $\alpha < d$. This completes our main task of satisfying the principal requirement of Lemma 2.1.

All that remains is to establish the incremental rate of growth condition, target recursion (2.1(a)). Since Lemma 5.1 gives an incremental rate of growth "by two's" for ρ_g, we define the function ψ, where $\psi(n) = \rho_g(2n)$, and show first that ψ satisfies (2.1(a)) in addition to the main target recursion (2.1(b)). We note that ψ is just ρ_g restricted to the even integers. Exchanging $2k$ for k in Inequality (6.3) yields, in terms of ψ,

$$m^{d-\alpha}\psi(k) - c_6 m^{d-\alpha}\alpha^{d/(d+1)}(2k)^{(d-\alpha)/(d+1)} \le \psi(m^d k), \tag{6.4}$$

so ψ satisfies the second target recursion with the slightly modified remainder function $r_\alpha(k) = 2^{(d-\alpha)/(d+1)}c_6\alpha^{d/(d+1)}k^{(\alpha-d)/d(d+1)}$. We next write the incremental growth inequality of Lemma 5.1 as

$$\rho_g(2n+2) \le \rho_g(2n) + c_5(2n+1)^{-\alpha/d}, \tag{6.6}$$

or in terms of ψ,

$$\psi(n+1) \le \psi(n) + 2^{-\alpha/d}c_5 n^{-\alpha/d}. \tag{6.7}$$

Using Lemma 2.1, this proves that

$$\psi(n) \sim \hat{\beta}_g n^{(d-\alpha)/d} \tag{6.8}$$

as $n \to \infty$ for some constant $\hat{\beta}_g = \hat{\beta}_g(d, \alpha)$.

Finally, to prove Theorem 1 for the function ρ_g, all we need to do is apply Lemma 5.2. Let $\{x_1, x_2, \ldots, x_{2n}\} \subset [0, 1]^d$ be a set of points with a greedy matching that achieves a total weight greater than or equal to $\rho_g(2n) - \epsilon$, where $\epsilon > 0$. If we add an additional point and form a greedy matching M of the $2n + 1$ points, then from Lemma 5.2 and the definition of $\rho_g(2n + 1)$, we have that

$$\rho_g(2n) - \epsilon - d^{\alpha/2} \le L_\alpha(M) \le \rho_g(2n + 1). \tag{6.9}$$

Also, since $\epsilon > 0$ and since $2n$ is even, Inequality (5.2) of Lemma 5.1 tells us that

$$\rho_g(2n) - d^{\alpha/2} \le \rho_g(2n + 1) \le \rho_g(2n). \tag{6.10}$$

Applying the relation (6.8) to $\rho_g(2n)$ and letting n go to infinity proves that $\rho_g(n) \sim \beta_g n^{(d-\alpha)/d}$ as $n \to \infty$, concluding the proof of Theorem 1. ■

A final issue that needs to be resolved is a comparison of the quantities $\hat{\rho}_g(n)$ and $\rho_g(n)$. By using a special tie-breaking perturbation of point sets of $[0, 1]^d$, the next three sections prove Theorem 2, which tells us that, remarkably, the sequences $\hat{\rho}_g(n)$ and $\rho_g(n)$ are identical.

7. BLOCKS AND LEGAL SHUFFLINGS OF EDGES

One obvious benefit of our analysis of ρ_g in the preceding sections is that it gives us the exact asymptotic weight of the worst possible greedy matching of n points in the unit d-cube. If we recall the definition

$$\rho_g(n) = \sup_{\substack{V_n \subset [0,1]^d \\ |V_n| = n}} \max_G \{L_\alpha(G) : G \text{ is a greedy matching of } V_n\}, \tag{7.1}$$

we see that $\rho_g(n)$ represents a supremum over all possible point sets of a greedy matching of n points that is formed by using a "worst possible" list of candidate edges for processing by the greedy algorithm. In other words, ρ_g can be associated with an algorithm that always yields a maximum-weight greedy matching of a given set of points.

In practice, this may not be the case since ties among lengths of edges in the candidate list are usually resolved in an arbitrary or random manner. Let us suppose, then, that we instead use an *optimal* greedy algorithm, i.e., one that resolves ties such that we always obtain a minimum-weight greedy matching. A natural question to ask is, What is the worst-case performance of this algorithm over all n-sets? The definition of $\hat{\rho}_g(n)$ embodies this scenario by replacing the inner maximum in the definition of $\rho_g(n)$ with a minimum.

The next lemma will be useful in resolving this issue. It guarantees that we can perturb any point set having a greedy matching M to obtain a point set with a greedy matching that is virtually identical to M yet exhibits no tied edges, as is expressed by the properties (i), (ii), and (iii) below. Despite the intuitive simplicity of these properties, the proof of the lemma is much more delicate than we would expect at the outset.

Lemma 7.1. *For any* $\epsilon > 0$ *and any greedy matching* M *of* $S = \{x_1, x_2, \ldots, x_n\} \subset [0,1]^d$, *there exists a point set* $S' = \{x'_1, x'_2, \ldots, x'_n\} \subset [0,1]^d$ *with an ordering* σ' *of the edges of the complete graph on* S' *such that, if the greedy algorithm applied to* σ' *forms the matching* M', *then*

(*i*) *The edge* $\{x'_i, x'_j\} \in M'$ *if and only if the edge* $\{x_i, x_j\} \in M$;
(*ii*) $L_\alpha(M') = \min_G\{L_\alpha(G) : G$ *is a greedy matching of* $S'\}$;
and
(*iii*) $0 \le L_\alpha(M) - L_\alpha(M') < \epsilon$.

Proof. Let $K = (S, E)$ be the complete graph on the point set S, and let σ be an ordering of the edges of E that gives rise to the greedy matching M. There is a k such that we can partition σ into k contiguous sublists or *blocks* $\sigma_1, \sigma_2, \ldots, \sigma_k$ such that, for any edges $e \in \sigma_i$ and $e' \in \sigma_j$, where $1 \le i \le k$ and $1 \le j \le k$,

$$\begin{aligned} |e| &< |e'|, && \text{if } i < j; \\ |e| &= |e'|, && \text{if } i = j; \text{ and} \\ |e| &> |e'|, && \text{if } i > j. \end{aligned} \tag{7.2}$$

Thus, we have a block for each distinct Euclidean edge length of the edge set E, and, within any block, all edge lengths are equal. In each block, we mark the edges that belong to the greedy matching M and say that the edges not belonging to M are unmarked. Next, for each i such that $1 \le i \le k$, we label the marked edges of the block σ_i in the order in which they appear in the block as $e_{i1}, e_{i2}, \ldots, e_{im(\sigma_i)}$, where $m(\sigma_i)$ is defined to be the number of marked edges

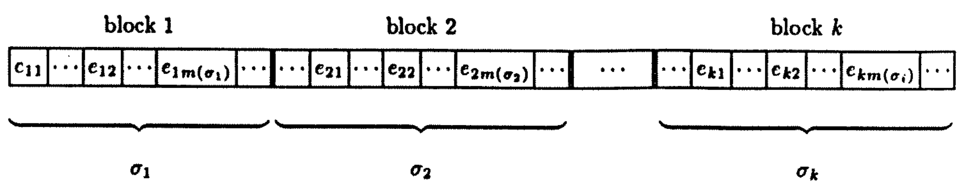

FIG. 1. The ordering σ and its marked edges.

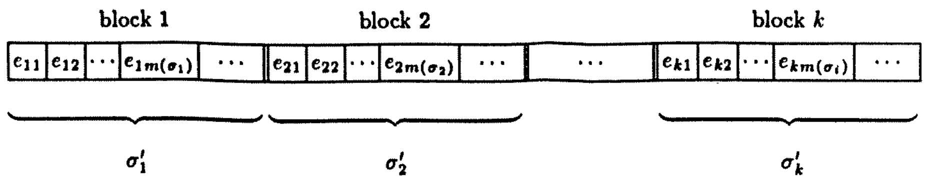

FIG. 2. The ordering σ'; marked edges of σ appear first in each block.

of σ_i; in other words,

$$m(\sigma_i) = |\{e : e \in \sigma_i \text{ and } e \in M\}|. \tag{7.3}$$

Figure 1 illustrates the ordering σ and its marked edges.

We now investigate ways of reordering the edges of σ to form a new ordering σ' without affecting the matching obtained when processing these lists of candidate edges with the greedy algorithm. First, define σ_i' to be the ordering $e_{i1}, e_{i2}, \ldots, e_{im(\sigma_i)}$ followed by the set of unmarked edges of σ_i, where the latter set is arranged in an arbitrary order as shown in Figure 2.

Now, all the marked edges that preceed a given unmarked edge e of σ still precede e in the new ordering $\sigma' = \sigma_1', \sigma_2', \ldots, \sigma_k'$, and since the relative orders of the marked edges in σ and σ' are identical, the matching obtained by applying the greedy algorithm to σ' will contain only marked edges. This means that the greedy algorithm applied to σ' will still form the matching M. This fact remains true for any permutation of the set of unmarked edges of each σ_i, where $1 \leq i \leq k$. In addition, it remains true even if we reassign the weights of the marked edges in such a way that the order of σ' is retained without violating the nondecreasing order of the edge weights. This fact will prove to be useful in the next section, in which we investigate a perturbation that satisfies the three properties of Lemma 7.1.

8. A PERTURBATION THAT BREAKS GREEDY TIES

We now set out to perturb the point set S to form the point set $S' = \{x_1', x_2', \ldots, x_n'\}$, where, for $1 \leq i \leq n$, the point $x_i' \in S'$ is the point $x_i \in S$ after the perturbation. For any edge $e = \{x_i, x_j\}$ of σ, we define $e' = \{x_i', x_j'\}$. We also let the *gap* $\Delta_{i,i+1}$ be the absolute difference between the Euclidean edge lengths corresponding to the two adjacent blocks σ_i and σ_{i+1}, i.e., if $e \in \sigma_i$ and $\tilde{e} \in \sigma_{i+1}$, then for $1 \leq i \leq k-1$,

$$\Delta_{i,i+1} = |\tilde{e}| - |e| > 0, \tag{8.1}$$

and, if $\in \sigma_1$, we define

$$\Delta_{0,1} = |e| > 0. \tag{8.2}$$

We then define Δ to be the *minimum gap* by setting

$$\Delta = \min_{0 \le i \le k-1} \Delta_{i,i+1}. \tag{8.3}$$

Note that, by the definition of $\Delta_{0,1}$, the minimum gap Δ is no larger than a minimum-length edge of σ.

We are now ready to describe the perturbation. Consider the block σ'_i of the order σ'. For each marked edge e_{ij}, where $1 \le j \le m(\sigma_i)$, for notational convenience we relabel the endpoints of e_{ij} so that $e_{ij} = \{y_j, z_j\}$. We then perturb the points z_j according to the following rule:

For all j satisfying $1 \le j \le m(\sigma_i)$, move z_j toward y_j a distance of

$$\delta_{ij} = \left(1 - \frac{j}{m(\sigma_i)+1}\right)\delta(\epsilon), \tag{8.4}$$

where the parameter $\delta(\epsilon)$ satisfies $0 < \delta(\epsilon) < \Delta$ and is to be chosen later.

We can note that our entire perturbed point set $S' = \{x'_1, x'_2, \ldots, x'_n\}$ is contained in the unit d-cube since the perturbation is accomplished by moving points a strictly positive distance along the marked edges of the complete graph on S, each of which is contained entirely in $[0, 1]^d$. We will momentarily see that no point moves too far.

We now can check that performing this perturbation scheme for all i, where $1 \le i \le k$, will guarantee properties (i)–(iii) of our lemma. First, to prove properties (i) and (ii), we note that as j ranges from 1 to $m(\sigma_i)$, the quantity $1 - j/(m(\sigma_i)+1)$ ranges from $m(\sigma_i)/(m(\sigma_i)+1)$ to $1/(m(\sigma_i)+1)$; hence, it is always strictly positive and less than one. Since we multiply this quantity by $\delta(\epsilon)$, which is less than the minimum gap Δ, we are assured that (1) for any edge $e_{ij} \in \sigma_i$, the Euclidean length $|e'_{ij}|$ remains greater than the length, after the perturbation, of any edge belonging to the preceding block σ_{i-1}; and (2) since Δ is at least as small as an edge of minimum length, the distance moved by the point z_j never exceeds the length of the edge $|e_{ij}|$. This verifies that $S' \subset [0, 1]^d$. Furthermore, since the reduction in edge length accomplished by the perturbation decreases as j increases, we see that we have retained the order of σ'_i; in other words, if we label the edges of the ordering σ' as $e_1, e_2, \ldots, e_u$, where $u = \binom{n}{2}$, then ordering the edges of the complete graph on the perturbed point set S' can produce the list $e'_1, e'_2, \ldots, e'_u$.

Consider now the formation of the matching M' by applying the greedy algorithm to the perturbed set S'. As the algorithm proceeds, it first considers the marked candidate edges $e'_{11}, e'_{12}, \ldots, e'_{1m(\sigma_1)}$ of block one. We claim that M' will contain only the marked edges of block one. To prove this claim, observe that changing the length of the edge e_{1j} by perturbing one of its endpoints can also change the length of only the remaining $n-2$ candidate edges that are incident with the perturbed point. This is of no consequence, however, since the order of marked edges has been retained by the perturbation and since the inclusion of the edge e'_{1j} into the matching M' forces all other edges incident with the endpoints of e'_{1j} to be ineligible for inclusion in M' for the remainder of the algorithm. Using an identical argument inductively on the k blocks proves that the edge e' is in the matching M' if and only if $e \in M$, which is Property (i).

In the preceding section, we noted that the ordering σ' produced the matching M regardless of the order of the set of unmarked edges of the σ'_i, where $1 \leq i \leq k$. Since our perturbation retains the order of marked edges of σ', we see that this property is preserved. In other words, arbitrarily reordering the unmarked edges of each σ'_i after carrying out the perturbation still produces the matching M'. Since ordering the edges of the complete graph on the perturbed point set S' can produce a list with ties occurring only among unmarked edges (we have perturbed the length of each marked edge so that no ties amongst the marked edges remain), we see that the total weight of *any* greedy matching of S' equals $L_\alpha(M')$. Property (ii) is a consequence of this fact.

To show that Property (iii) is satisfied, we represent the difference in total weights of M and M' by

$$L_\alpha(M) - L_\alpha(M') = \sum_{i=1}^{k} \sum_{j=1}^{m(\sigma_i)} \{|e_{ij}|^\alpha - |e'_{ij}|^\alpha\}; \tag{8.5}$$

The first sum is over blocks, and the second is over marked edges of the i'th block. That the perturbation parameter $\delta(\epsilon)$ can be chosen so that Property (iii) holds is immediate from this representation because, as $\delta(\epsilon) \to 0$, we have that $|e'_{ij}| \to |e_{ij}|$. For an explicit choice of $\delta(\epsilon)$ for a given ϵ that satisfies Property (iii) of Lemma 7.1, one can consult Snyder [11].

This concludes the proof of Lemma 7.1, which makes brief the proof of Theorem 2 in the next section. ∎

9. A WORST-CASE EQUIVALENCE OF GREEDY MATCHINGS

We now commence with the proof of Theorem 2, which is made simple by Lemma 7.1. We first define $S = \{x_1, x_2, \ldots, x_n\}$ to be a set of n points in $[0, 1]^d$ having a greedy matching M of total weight $L_\alpha(M) \geq \rho_g(n) - \epsilon$, for $\epsilon > 0$. Let S' be a perturbation of S, and let σ' and M' be an ordering of the edges of the complete graph on S' and the greedy matching obtained by using this ordering, respectively, such that properties (i)–(iii) of Lemma 7.1 are satisfied.

For convenience, we recall the definition of $\hat{\rho}_g(n)$:

$$\hat{\rho}_g(n) = \sup_{\substack{V_n \subset [0,1]^d \\ |V_n| = n}} \min_G \{L_\alpha(g) : G \text{ is a greedy matching of } V_n\}. \tag{9.1}$$

Now, by Property (ii) of Lemma 7.1, by S' we have exhibited a set of n points in $[0, 1]^d$ whose minimum-weight greedy matching is of total weight $L_\alpha(M')$, so, by the definition of $\hat{\rho}_g(n)$, we have that $L_\alpha(M') \leq \hat{\rho}_g(n)$. Since M and M' satisfy Property (iii) of Lemma 7.1, we are guaranteed that

$$0 \leq L_\alpha(M) - L_\alpha(M') \leq \rho_g(n) - \epsilon - L_\alpha(M') < \epsilon; \tag{9.2}$$

hence, we obtain

$$\rho_g(n) - 2\epsilon < \hat{\rho}_g(n) \tag{9.3}$$

for all $n \geq 1$. Since this inequality is true for all $\epsilon > 0$ and since $\hat{\rho}_g(n)$ is majorized by $\rho_g(n)$, the two sequences must be equal for all $n \geq 1$, and the theorem is proved. ■

10. CONCLUSIONS

The asymptotic result given in Theorem 1 combined with the result of Theorem 2 raises some interesting issues concerning the effectiveness of the greedy heuristic for matchings. First recall the bound of Reingold and Tarjan [10], namely, if $L(\mathrm{GM})$ and $L(\mathrm{M}^*)$ are the weights of a greedy matching and a minimal matching of n points in the plane, respectively, where n is even, then

$$\frac{L(\mathrm{GM})}{L(\mathrm{M}^*)} \leq \frac{4}{3} n^{\log_2 1.5} - 1. \tag{10.1}$$

In view of (10.1) it is natural to suspect that the greedy algorithm is grossly inefficient. Perhaps we now have reasons to reconsider.

We note first that in addition to establishing the bound of Inequality (10.1), Reingold and Tarjan [10] constructed a collinear configuration of points that attained the worst-case ratio expressed by the right-hand side of (10.1). This example point configuration also gives us the ratio of Inequality (10.1) if we pit the worst-case greedy matching algorithm that we have associated with ρ_g against the optimal greedy algorithm that we have associated with $\hat{\rho}_g$. Theorem 2 tells us that the discrepancy expressed by respectively exchanging $\rho_g(n)$ and $\hat{\rho}_g(n)$ for $L(\mathrm{GM})$ and $L(\mathrm{M}^*)$ in (10.1) does not exist in the worst case.

We also note that the following theorem of Snyder [11], which is the analog of Theorem 1 for minimal matchings, holds for the sequence

$$\rho_m(n) = \sup_{\substack{V_n \subset [0,1]^d \\ |V_n| = n}} \min_M \left\{ \sum_{e \in M} |e|^\alpha : M \text{ is a matching of } V_n \right\}. \tag{10.2}$$

Theorem. *there exists a constant β_m depending on the dimension $d \geq 2$ and the edge-weighting function $w(e) = |e|^\alpha$, where $0 < \alpha < d$, such that as $n \to \infty$,*

$$\rho_m(n) \sim \beta_m n^{(d-\alpha)/d}. \tag{10.3}$$

Using Theorem 1 and this theorem along with bounds on the weights of worst-case minimal and greedy matchings in Avis [1], Supowit et al. [13], and Iri et al. [7], one can observe that the ratio of worst-case greedy to worst-case minimal matchings is a constant as $n \to \infty$, and this constant is at most two Snyder [11].

Hence, in the worst case, in which our matchings are of sufficiently great weight, the greedy algorithm turns out to be equivalent to any minimal matching algorithm, up to a constant factor. This is a reflection of the fact that for Euclidean matchings in bounded regions such as the unit cube, the example of Reingold and Tarjan [10] has a greedy matching that is bounded in weight and a minimal matching whose weight goes to zero as $n \to \infty$. In other words, the example of Reingold and Tarjan is not necessarily one for which the greedy algorithm performs poorly, but it is one for which the weight of a minimal matching is unusually small.

For this reason, Avis [1] has argued that a more reasonable measure of relative performance of matching algorithms in the context of applications such as mechanical plotters is the absolute difference $L(\mathrm{GM}) - L(\mathrm{M}^*)$, since this quantity is proportional to the wasted movement of the pen while it is in the "up" position.

Our worst-case scenarios and the example of Reingold and Tarjan [10] represent different ends of the spectrum in terms of the total weight of the heuristic matching obtained using the greedy algorithm. Because of the similarity of our worst-case growth rate to the probabilistic growth rate of Avis et al. [3], it appears that, except for under unusual circumstances, we can expect the greedy algorithm to perform relatively well. It is an open problem to determine exactly what these circumstances are.

Another open problem that has received a great deal of attention is the determination of constants like β_g and β_m. Although our methods do not lead us down this path, we do note that Theorem 2 gives us a concise necessary condition for a worst-case point configuration for the optimal greedy algorithm: If any set of n points attains the worst-case weight of a greedy matching using an optimal greedy algorithm, then it must produce greedy matchings of identical weight, regardless of how ties are broken. This easily investigated condition may be of assistance in determining worst-case arrangements of points.

ACKNOWLEDGMENT

The exposition in this paper has profited from the assistance of an unusually careful referee. The proof of Lemma 4.2 is based on a suggestion of the referee, shortening a proof that appeared in an earlier version.

This work was supported in part by grants NSF DMS88-12868, NSA MDA 904-89-H-2034, AFOSR-89-0301, DAAL03-89-G-0092, and Georgetown University 1989 Summer Academic Grant.

REFERENCES

[1] D. Avis, Worst case bounds for the Euclidean matching problem. *Comput. Math. Appl.* **7** (1981) 251–257.

[2] D. Avis, A survey of heuristics for the weighted matching problem. *Networks* **13** (1983) 475–493.

[3] D. Avis, B. Davis, and J. M. Steele, Probabilistic analysis of a greedy heuristic for Euclidean matching. *Probability Eng. Informational Sci.* **2** (1988) 143–156.

[4] N. Christofides, Worst case analysis of a new heuristic for the traveling salesman problem. Technical Report, G.S.I.A., Carnegie-Mellon University, Pittsburgh, PA (1976).

[5] J. Edmonds, Paths, trees, and flowers. *Can. J. Math.* **17** (1965) 449–467.

[6] E. Gabow, An efficient implementation of Edmonds' algorithm for maximum matching on graphs. *J. Assoc. Comp. Mach.* **23**, (1976) 221–234.

[7] M. Iri, K. Murota, and S. Matsui, Heuristics for planar minimum-weight perfect matchings. *Networks* **13** (1983) 67–92.

[8] M. Iri and A. Taguchi, The determination of the pen movement of an *XY*-plotter and its computational complexity (in Japanese). *Proceedings of the Spring Conference of the O.R. Society of Japan*, **P-8** (1980) 204–205.

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

[10] E. M. Reingold and R. E. Tarjan, On a greedy heuristic for complete matching. *SIAM J. Comput.* **10** (1981) 676–681.

[11] T. L. Snyder, Asymptotic worst case lengths in some problems from classical computational geometry and combinatorial optimization. Ph.D. Thesis, Princeton University, Princeton, NJ (1987).

[12] J. M. Steele and T. L. Snyder, Worst case growth rates of some classical problems of combinatorial optimization. *SIAM J. Comput.* **18** (1989) 278–287.

[13] K. J. Supowit, E. M. Reingold, and D. A. Plaisted, The traveling salesman problem and minimum matchings in the unit square. *SIAM J. Comput.* **12** (1983) 144–156.

Received May 1988
Accepted December 1989