On Optimal Guillotine Partitions Approximating Optimal d-Box Partitions

This research was supported in part by the National Science Foundation under Grant DCR - 8503163

Teofilo F. Gonzalez, Mohammadreza Razzazi,
Department of Computer Science
University of California
Santa Barbara, CA 93106

Man-Tak Shing,
Department of Computer Science
Naval Postgraduate School
Monterey, CA 93943

and

Si-Qing Zheng
Department of Computer Science
Louisiana State University
Baton Rouge, LA 70803-4020

Abstract: Given a set of \( n \) points, \( P \), in \( E^d \) (the plane when \( d = 2 \)) that lie inside a \( d \)-box (rectangle when \( d = 2 \)) \( R \), we study the problem of partitioning \( R \) into \( d \)-boxes by introducing a set of orthogonal hyperplane segments (line segments when \( d = 2 \)) whose total \( (d - 1) \)-volume (length
when \( d = 2 \) is the least possible. In a valid \( d \)-box partition, each point in \( P \) must be included in at least one partitioning orthogonal hyperplane segment. Since this minimization problem is NP-hard and thus likely to be computationally intractable, we present an approximation algorithm to generate a suboptimal solution. This solution is obtained by finding an optimal guillotine partition, i.e., a special type of rectangular partition, which can be generated in \( O(dn^{2d+1}) \) time. We present a simple proof that the \((d-1)\)-volume of an optimal guillotine partition is not greater than \( 2d - 4 + 4/d \) times the \((d - 1)\)-volume of an optimal \( d \)-box partition.

**Keywords:** Approximation algorithms, \( d \)-box partitions, polynomial time complexity, guillotine partitions, computational geometry.
1. Introduction.

Given a set of points, \( P \), in \( E^2 \) that lie inside a rectangle \( R \), we study the problem of partitioning \( R \) into rectangles by introducing a set of orthogonal line segments whose total length is the least possible, and each point in \( P \) is included in at least one of the partitioning line segments. We shall refer to this problem as the \( RG-P_2 \) problem, where \( RG \) stands for rectangle, \( P \) for a set of points, and the two for 2-D. We use \( E(I) \) to represent a set of partitioning line segments in a feasible solution for \( I = (R, P) \). The \( RG-P_2 \) problem was shown to be NP-hard in [LPRS].

Approximation algorithms for the \( RG-P_2 \) problem appear in [GRZ], [GZ1], [GZ2], and [GZ3]. The currently best approximation algorithms for the \( RG-P_2 \) problem are summarized in Table I.

**TABLE I: Approximation algorithms for partitioning a rectangle**

In VLSI design, the problem of dividing routing regions into channels can be reduced to the \( RG-P_2 \) problem with interior holes instead of points ([R]). Several approximation algorithms for this more general problem exist (see [DC], [L], [Le1] and [Le2]). The algorithms with the smallest approximation bound are the ones reported in [Le1] and [Le2]. The algorithm given in [Le2] invokes the algorithm reported in [GZ1] for the \( RG-P_2 \) problem as a sub-procedure. The algorithms in [GZ1], [GZ3], and [GRZ] generate guillotine partitions, but the ones in [GZ3], and [GRZ] are in the worst case "closer" to optimal. Therefore, a smaller approximation bound for the general problem can be obtained by using these algorithms in the procedure reported in [Le2].

Let \( R \) be a rectangle and let \( P \) be a set of points inside \( R \). A rectangular partition \( E(I) \), for \( I = (R, P) \), has a guillotine cut if there is a line segment in \( E(I) \) that partitions the rectangular boundary \( R \) into two rectangles. We say that a rectangular partition \( E(I) \) is a guillotine partition if either \( E(I) \) is empty or \( E(I) \) has a guillotine cut that partitions \( R \) into \( R_1 \) and \( R_2 \), and both \( E(I_1) \) (edges from \( E(I) \) in \( R_1 \)) and \( E(I_2) \) (edges from \( E(I) \) in \( R_2 \)) are guillotine partitions for \( I_1 = (R_1, P_1) \) and \( I_2 = (R_2, P_2) \), respectively. The minimum edge length guillotine partitioning problem consists of finding a guillotine partition of least total length, i.e., the sum of the lengths of the edges in the partition. It is simple to see that any guillotine partition is a rectangular partition, but the converse is not necessarily true (see Figure 1). Du et al. [DPS] studied the problem of finding an optimal guillotine partition for any instance \( I = (R, P) \) of the \( RG-P_2 \) problem. They showed that such a partition can be found by dynamic programming in \( O(n^5) \) time, where \( n \) is the total number of points. Du et al. [DPS] also showed that the length of an optimal guillotine partition is at most twice the length of an optimal rectangular partition for the \( RG-P_2 \) problem. Their proof of this bound is lengthy and complex. Our proof, which can be easily generalized to the problem defined over an arbitrary number of dimensions, is simpler. A complex proof showing that for 2-D the length of an optimal guillotine partition is within 1.75 times the length of an optimal rectangular partition is given in [GZ3], file 1.1.

Figure 1: Rectangular and guillotine partitions.

We also consider a more general version of \( RG-P_2 \), denoted by \( RG-P_d \), which is defined over \( E^d \). Specifically, given a set \( P \) of points in \( E^d \) that lie inside a \( d \)-box \( R \), we study the problem of partitioning \( R \) into \( d \)-boxes by introducing a set of orthogonal hyperplane segments whose total \((d-1)\)-volume is the least possible. In a valid partition, each point in \( P \) must be included in at least one partitioning orthogonal hyperplane segment. In what follows, when we refer to a guillotine cut (partition) we mean the obvious \( d \)-dimensional generalization of a guillotine cut (partition). \(^1\)

---

\(^1\)By a \( d \)-box (in \( E^d \) space) we mean a \( d \)-dimensional rectangle, whereas by hyperplane we mean a \((d-1)\)-
Gonzalez, Razzaqi and Zheng [GRZ] present an $O(dn \log n)$ algorithm that generates solutions that are within $2d$ of optimal for the RG-$P_d$ problem. In section 2, we present a simple proof that the $(d-1)$-volume of an optimal guillotine partition is not greater than $2d - 4 + 4/d$ times the $(d-1)$-volume of an optimal $d$-box partition. In theorem 2, we show that an optimal guillotine partition can be obtained via dynamic programming in $O(dn^{2d+1})$ time, where $n$ is the number of points in $P$. Therefore, an optimal guillotine partition provides a polynomial time approximation to the $d$-box partition problem when $d$ is bounded by some fixed constant. It is worthwhile mentioning that the algorithm in this paper always generates solutions that are not farther from optimal than the ones generated by the algorithm in [GRZ], since both algorithms generate guillotine partitions, and our algorithm generates an optimal guillotine partition. However, the algorithm in [GRZ] is always faster.

For more than two dimensions, our problem is far removed from its original application; however, it has other applications when $d = 3$. For example, suppose we have a solid block of material and we know exactly the location of all its "impure" points. Our problem consists of partitioning the block into sub-blocks without impure material inside them in such a way that the total area of the cuts is the least possible. This is in general a good estimate for the cost of partitioning the block into sub-blocks. The objective function can be modified to include the more traditional setup cost to align the machine in the various directions. Our dynamic programming algorithm can be easily modified so that it finds optimal guillotine partitions with respect to this criteria. However, it is not clear whether or not this type of partition would also be a good approximation for the corresponding $d$-box partition problem.

2. Bounds for $d$-Dimensional Guillotine Partitions.

An instance $I = (R=(a, X), P)$ of the RG-$P_d$ problem consists of a $d$-box $R$ (where $a = (a_1, a_2, ..., a_d)$ is the "lower-left" corner of the boundary $R$ (origin of $R$), and $X = (X_1, X_2, ..., X_d)$ are the dimensions or sizes of $R$) and a set of points $P = \{ p_1, p_2, ..., p_n \}$ inside $R$. Let $E(I)$ denote the set of orthogonal hyperplane segments in a valid $d$-box partition of $I$, and let $G(I)$ denote the set of orthogonal hyperplane segments in a minimum $(d-1)$-volume guillotine partition. We show that $V(G(I)) \leq 2d - 4 + 4/d \cdot V(E(I))$, where $V(S)$ represents the sum of the $(d-1)$-volume of the segments in set $S$. To prove our bound we define, via procedure TRANSF_TO_G, a set of orthogonal hyperplane segments $E'(I)$ such that $E'(I) E(I)$ forms a guillotine partition for $I$ and $V(E'(I)) \leq 2d - 4 + 4/d \cdot V(E(I))$.

In what follows we refer to the dimensions of $E_d$ by the integers $1, 2, ..., d$ (in 2-space we have the first dimension or $1^{st}$-axis, and the second dimension or $2^{nd}$-axis). Let $E_j(I)$ be the set of hyperplane segments orthogonal to axis $j$ in $E(I)$. An orthogonal hyperplane segment to the $j^{th}$-axis that partitions the $d$-box $R$ into two $d$-boxes is called a $j^{th}$-axis orthogonal cut. A $d$-box partition $E(I)$ has a half $j^{th}$-axis overlapping cut $i$ if $i$ is a $j^{th}$-axis orthogonal cut such that $V(l E(i)) = 0.5 \Pi_{l=y} X_y$. We say that $E(I)$ has a $j^{th}$-axis guillotine cut, $i$, if $i$ is a $j^{th}$-axis orthogonal cut such that $V(l E(i)) = \Pi_{l=y} X_y$. Suppose $R$ is partitioned by a $j^{th}$-axis orthogonal cut into two $d$-boxes, $R_1$ and $R_2$. With respect to such a partition we define $E(I_1)$ and $E(I_2)$ as the set of hyperplane segments $E(I)$ inside $R_1$ and $R_2$, respectively.

Assume that $E(I)$ is non-empty, and assume without loss of generality that the $d$-box has been rotated so that $V(E_j(I)) \leq V(E_d(I))$ for all $j < d$. As we shall see later on, this is a very dimensional plane. Whenever we refer to a hyperplane segment, we mean a $(d-1)$-dimensional $d$-box.
useful property in the proof of theorem 1. It is very important to remember that dimension \( d \) will remain dimension \( d \) throughout the transformation process. We use \( E'(I) \) to denote all of the portions of the segments introduced by procedure \( TRANS_E.TO_G \) that do not overlap with the segments in \( E(I) \). Initially \( E'(I) \) is empty and none of the facets of \( R \) are colored. When a facet of \( R \) is colored it indicates that procedure \( TRANS_E.TO_G \) has introduced a hyperplane segment (or part of it) whose \((d-1)\)-volume is equal to the \((d-1)\)-volume of this facet, and which has not yet been accounted for by hyperplane segments in \( E(I) \). \footnote{By facets of \( R \) we mean only the \( 2d \) outer facets of \( R \), and not the facets of the \( d \)-boxes within \( R \).} Procedure \( TRANS_E.TO_G \), which we formally define below, first checks if there is a \( d \)-overlapping cut and if so, it introduces a \( d^{th} \)-axis orthogonal cut along the half \( d^{th} \)-axis overlapping cut. As a result of introducing this cut two subinstances are generated and a hyperplane segment is added to \( E'(I) \). It is simple to verify that the \((d-1)\)-volume of the newly introduced segment in \( E'(I) \) is at most equal to the \((d-1)\)-volume of the segments in \( E(I) \) that overlap with the \( d^{th} \)-axis orthogonal cut. Such a segment is not part of \( E(I_1) \) or \( E(I_2) \). The facets of \( R_1 \) and \( R_2 \) that were parts of colored facets of \( R \) remain colored. At this point we invoke procedure \( TRANS_E.TO_G \) recursively on the two resulting subinstances. If there is no \( d^{th} \)-axis overlapping cut, \( TRANS_E.TO_G \) checks if there is a \( 1^{st} \)-axis guillotine cut, and if so, it partitions the rectangle through one such cut and at most two subinstances are generated. All of the facets of \( R_1 \) and \( R_2 \) that were parts of colored facets of \( R \) lose their coloring at this point. The reason for this is that the \((d-1)\)-volume of the \( 1^{st} \)-axis guillotine cut will account for the \((d-1)\)-volume of the colored facets of \( R \). We apply procedure \( TRANS_E.TO_G \) recursively to each non-empty subinstance. If there is no \( 1^{st} \)-axis guillotine cut, \( TRANS_E.TO_G \) introduces a \( 1^{st} \)-axis orthogonal cut, called a mid-cut, that intersects the center of the \( d \)-box (i.e., it includes the point \((o_1 + X_1/2, o_2 + X_2/2, \ldots, o_d + X_d/2)\)). As we shall prove later on, when a mid-cut is introduced each of the two resulting subinstances has at least one segment inside it which is not orthogonal to the \( d \)-axis. Suppose that \( R_1 \) contains point \( o \) of \( R \). The facets in \( R_1 \) and \( R_2 \) that were colored in \( R \) remain colored. The facet in \( R_1 \) orthogonal to the \( 1^{st} \)-axis which does not include point \( o \) gets colored at this point. The \((d-1)\)-volume of such facet is equal to the \((d-1)\)-volume of the mid-cut just introduced. The procedure is applied recursively to the resulting subinstances. The procedure is formally defined below. The first invocation involves an uncolored \( d \)-box partition \( E(I) \) of \( R \).

\begin{verbatim}
procedure TRANS_E.TO_G(I=(o, X), E(I), d)
begin       Relabel the dimensions so that X_1 X_2 ... X_{d-1};
    /* Note that dimension d is never relabeled. Extreme care must be taken after this step    since the axes have been relabeled. For clarity we do not include all the details needed    because of the relabeling. */
    /* In what follows we partition R into R_1 and R_2 by introducing a hyperplane segment    along a cut. We assume that R_1 contains point o of R. */
    case      : E(I) has a half d^{th}-axis overlapping cut:
        Introduce a hyperplane segment in R along one such cut;
        The facets of R_1 and R_2 which were colored in R remain colored (even if sliced);
\end{verbatim}
: $E(I)$ has an $1^{st}$-axis guillotine cut:

Introduce a hyperplane segment in $R$ along one such cut;

Delete all colors from the facets of $R_1$ and $R_2$;

: else:

Introduce into $R$ a $1^{st}$-axis orthogonal cut that intersects the center of $R$; /* mid-cut */

All the facets of $R_1$ and $R_2$ which were parts of colored facets of $R$
(even if sliced) remain colored;

Color the facet of $R_1$ orthogonal to the $1^{st}$-axis which does not include point $o$;

endcase

if $E(I_1)$ then invoke $TRANS_E.TO_G$ recursively with $(I_1, E(I_1), d)$;

if $E(I_2)$ then invoke $TRANS_E.TO_G$ recursively with $(I_2, E(I_2), d)$;

end

It is important to note that procedure $TRANS_E.TO_G$ is only used to establish our approximation bound. Let $E'(I)$ be the set of hyperplane segments introduced by procedure $TRANS_E.TO_G$ that do not overlap with the segments in $E(I)$. Clearly, $E(I) E'(I)$ is a guillotine partition. Remember that $E_j(I)$ represents the set of hyperplanes in $E(I)$ orthogonal to axis $j$. We define $E_j(I)$ similarly, but with respect to $E'(I)$. We define $X$ as $\prod_{i=1}^{d} X_i$, for $1 \leq j \leq d-1$. In what follows, we assume that $X_1 X_2 \ldots X_{d-1}$. A $d$-box $R = (o, X)$ is said to be of type $i$ ($1 \leq i \leq d-1$) if $i$ is the largest integer such that $X_1 X_2 X_i$ for all $I$, the type of a $d$-box is uniquely defined. Before proving our main result (theorem 1), we prove the following intermediate results (lemmas 1 and 2). It is important to remember that once $TRANS_E.TO_G$ begins, dimension $d$ will not be relabeled.

Lemma 1: Every invocation made to procedure $TRANS_E.TO_G$ $(I, E(I), d)$, satisfies the following conditions:

(a) $E(I)$ is non-empty, and

(b) if $E(I)$ has a $d^{th}$-axis guillotine cut then none of the facets of $R$ are colored; otherwise the only facets of $R$ which could be colored are those facets orthogonal to the $j^{th}$-axis which do not include point $a$, for $1 \leq j \leq i$, where $i$ is the type of $R$.

Proof: Initially, $E(I)$ and none of the facets of $R$ are colored. Therefore, the first invocation to procedure $TRANS_E.TO_G$ satisfies (a) and (b). We now show that if upon entrance to the procedure the conditions are satisfied, then the invocations made directly from it will also satisfy
(a) and (b). Assume \((I, E(I), d)\) satisfies conditions (a) and (b). There are three cases depending on the type of cut introduced by procedure \(TRANS_E_TO_G\).

**case 1:** Procedure \(TRANS_E_TO_G\) partitions \(R\) along a half \(d^{th}\)-axis overlapping cut.

Suppose that \(E(I)\) has a \(d^{th}\)-axis guillotine cut. From (b) we know that none of the facets of \(R\) are colored, and no facet will get colored at this step. Since each invocation made directly from the current invocation is on non-empty problem instances, it follows that invocations made directly from this call satisfy both properties (a and b). On the other hand, if \(E(I)\) does not have a \(d^{th}\)-axis guillotine cut, then the half \(d^{th}\)-axis overlapping cut cannot be a \(d^{th}\)-axis guillotine cut. Therefore, there is at least one hyperplane segment orthogonal to the \(j^{th}\)-axis, for some \(1 \leq j < d\), on each side of the half \(d^{th}\)-axis overlapping cut. Since a \(d^{th}\)-axis orthogonal cut over this half \(d^{th}\)-axis overlapping cut is introduced, the resulting two \(d\)-box partitions \((E(I_1)\) and \(E(I_2)\)) must be non-empty, otherwise \(E(I)\) is not a \(d\)-box partition. Clearly, \(R_1\) and \(R_2\) are of type \(i\) (since the first \(d-1\) dimensions of \(R_1\), \(R_2\) and \(R\) are identical), there are no \(d^{th}\)-axis guillotine cuts (since \(R\) had no guillotine cuts and a cut orthogonal to the \(d\)-axis was introduced) and the only facets of \(R_1\) and \(R_2\) which could be colored are those facets orthogonal to the \(j\)th-axis which do not include point \(o\), for \(1 \leq j < i\). Therefore, each invocation made directly from the current invocation satisfy both properties (a and b). This completes the proof for this case.

**case 2:** Procedure \(TRANS_E_TO_G\) partitions \(R\) along a \(1^{st}\)-axis guillotine cut.

Since just after partitioning \(R\) all of the facets in \(R_1\) and \(R_2\) lose their coloring, and we invoke procedure \(TRANS_E_TO_G\) only if their corresponding \(d\)-box partition is non-empty, then the invocations made in this case satisfy (a) and (b). This completes the proof of this case.

**case 3:** Procedure \(TRANS_E_TO_G\) introduces a mid-cut.

Since \(E(I)\) does not have a \(1^{st}\)-axis guillotine cut, because case 2 does not apply, then each of the resulting \(d\)-box partitions has at least one hyperplane segment orthogonal to the \(j^{th}\)-axis for some \(1 \leq j \leq d-1\). Since \(R\) is of type \(i\), and a mid-cut is introduced orthogonal to the \(1^{st}\)-axis, then the type of \(R_1\) and \(R_2\) is \(k\), for some integer \(k\geq i\). Clearly, the second subinstance satisfies (a) and (b). Since \(o\) is in \(R_1\), the facet orthogonal to the \(1^{st}\)-axis which does not include \(o\) is not colored in \(R_1\) when the mid-cut is introduced. Immediately after that step it gets colored. Therefore, the first subinstance also satisfies (a) and (b). This completes the proof for this case and the lemma.

**Lemma 2:** For any non-empty \(d\)-box partition \(E(I)\) of any instance \(I\) of the \(RG-P_d\) problem, procedure \(TRANS_E_TO_G\) generates a set \(E'(I)\) of \((d-1)\)-dimensional hyperplane segments such that \(V(E'_d(I)) = V(E_d(I))\), and \(\sum_{j=1}^{d-1} V(E'_j(I)) = (2d-3) \sum_{j=1}^{d-1} V(E_j(I))\).

**Proof:** First we show that \(V(E'_d(I)) = V(E_d(I))\). This is simple to prove because \(d^{th}\)-axis orthogonal cuts are only introduced over half \(d^{th}\)-axis overlapping cuts. Each time a \(d^{th}\)-axis orthogonal cut is introduced the segment \(l'\) added to \(E'_d(I)\) has \((d-1)\)-volume that is at most equal to the \((d-1)\)-volume of the segments in the half \(d^{th}\)-axis overlapping cut \(l\) in \(E_d(I)\). Since neither \(l\) nor \(l'\) are in the interior of the resulting subinstances after this operation, these hyperplane segments of \(E_d(I)\) are never charged more than once, and thus \(V(E'_d(I)) = V(E_d(I))\).

Let us now show that \(\sum_{j=1}^{d-1} V(E'_j(I)) = (2d-3) \sum_{j=1}^{d-1} V(E_j(I))\). It is simple to verify that procedure \(TRANS_E_TO_G\) does not color a facet more than once, and at the end none of the facets are colored. The only place where facets lose their color is when procedure \(TRANS_E_TO_G\) introduces a \(1^{st}\)-axis guillotine cut. Let us examine this case in more detail. Since \(R\) is of type
i, and the only facets which could be colored are those that do not include a and which are orthogonal to axis \( j \) for \( 1 \leq i < d \) (by the lemma 1), we know that the \((d-1)\)-volume which could be colored is at most \( \sum_{j=1}^{d-1} X \). The \((d-1)\)-volume of the \( 1^{st} \)-axis guillotine cut is \( X \). Since \( X \) of every facet colored by procedure \( TRANS_E \rightarrow TO_G \), and the \((d-1)\)-volume of a set of \( j^{th} \)-axis guillotine cuts identified by procedure \( TRANS_E \rightarrow TO_G \) is at most \((2d-3)\) times the \((d-1)\)-volume of the facets colored by procedure \( TRANS_E \rightarrow TO_G \), it then follows that \( \sum_{j=1}^{d-1} V(E_j(I)) \), This concludes the proof of the lemma.

Theorem 1: Let \( E_{opt}(I) \) be an optimal \( d \)-box partition for any instance \( I \) of the \( RG-P_d \) problem, and let \( G(I) \) be an optimal guillotine partition for \( I \). Then \( V(G(I)) (2d-4+4/d) V(E_{opt}(I)) \).

Proof: Let \( E(I) \) be any \( d \)-box partition for \( I \). Assume without loss of generality that \( V(E_d(I)) \) \( V(E_j(I)) \) for \( 1 \leq j < d \) (i.e., the sum of the \((d-1)\)-volume of the hyperplane segments orthogonal to the \( j^{th} \)-axis is larger than the ones orthogonal to the \( i^{th} \)-axis), as otherwise a simple initial relabeling is performed before we start the transformation. It is important to keep in mind that this relabeling is performed only once. Apply procedure \( TRANS_E \rightarrow TO_G \) to \( E(I) \). Combining the fact that \( V(E_d(I)) V(E_j(I)) \) for \( 1 \leq j < d \) together with lemma 2 and the assumption that \( d \geq 2 \), we know that

\[
V(E(I) E'(I)) (2d-2) \sum_{j=1}^{d-1} V(E_j(I)) + 2 V(E_d(I)).
\]

This function achieves its maximum value when \( V(E_j(I)) = V(E(I)) \) \( / d \), Therefore,

\[
V(E(I) E'(I)) ((2d-2)(d-1) + 2) V(E(I))/d = (2d-4+4/d) V(E(I)).
\]

Since \( E(I) E'(I) \) is a guillotine partition for \( I \), \( V(G(I)) V(E(I) E'(I)) \). Hence, \( V(G(I)) (2d-4+4/d) V(E(I)) \).

Theorem 2: An optimal guillotine partition for a \( RG-P_d \) problem may be constructed in \( O(d n^{2d+1}) \) time.

Proof: Let \( g(I_1, I_2, ..., I_d) \) be the \((d-1)\)-volume of an optimal guillotine partition for the \( d \)-box defined by the intervals \( I_1, I_2, ..., I_d \), where \( I_j = [l_j, r_j] \), \( l_j \leq r_j \), and \( l_j \) is the \( j^{th} \)-coordinate value of a point in \( P \). For any \( I_1, I_2, ..., I_d \), one can easily compute \( g(I_1, I_2, ..., I_d) \) recursively by trying all \( d n \) guillotine cuts (where \( n \) is the number of points in the \( d \)-box formed by \( I_1, I_2, ..., I_d \)) and then selecting the minimum one. Since there are \( O(n^{2d}) \) of such \( g \)'s that need to be computed in order to find the \((d-1)\)-volume of an optimal partition, then by using dynamic programming the overall time complexity is \( O(d n^{2d+1}) \). By recording all valid guillotine partitions while computing volumes, this procedure can be easily modified to construct an optimal guillotine partition, rather than only computing the optimal \((d-1)\)-volume.

3. Discussion.

For any \( d \geq 2 \), it is simple to find a problem instance \( I \) such that \( V(G(I)) \) is about \( 1.5 V(E_{opt}(I)) \) [DPS]. One of such problem instances is an obvious generalization of the \( d \)-box partition in Figure
2. Although we have obtained the upper bound $2d^{-4}4^{-1}/d$, it is not clear whether or not there are problem instances that achieve (asymptotically) this bound for $d > 2$.

![Figure 2.](image)

In our transformation procedure, $TRANS_E TO_G$, one may replace a $1^{st}$-axis guillotine cut by a $j^{th}$-axis guillotine cut ($1 j < d$). This has no effect in the worst case approximation bound obtained in this paper; however, most of the time it will introduce a set of hyperplane segments with smaller $(d-1)$-volume.

By using standard speed-up techniques [Y] it seems impossible to reduce the $O(dn^{2d+1})$ time complexity bound. It would be of theoretical interest to try to improve the time required to find an optimal guillotine partition.

4. References.


