International Journal of Computational Geometry & Applications

DECISION TREES FOR GEOMETRIC MODELS*

ESTHER M. ARKIN[†]
Department of Applied Mathematics and Statistics
State University of New York at Stony Brook
Stony Brook, NY 11794-3600, USA

and

HENK MEIJER[‡]
Department of Computer and Information Science
Queen's University
Kingston, Ontario, K7L 3N6, CANADA

and

JOSEPH S. B. MITCHELL[§]
Department of Applied Mathematics and Statistics
State University of New York at Stony Brook
Stony Brook, NY 11794-3600, USA

and

DAVID RAPPAPORT[¶]
Department of Computer and Information Science
Queen's University
Kingston, Ontario, K7L 3N6, CANADA

and

STEVEN S. SKIENA[‖]
Department of Computer Science
State University of New York at Stony Brook
Stony Brook, NY 11794-4400, USA

Received (received date)
Revised (revised date)
Communicated by Editor's name

---

*A preliminary version of this paper appeared in the *Proc. Ninth Annual ACM Symposium on Computational Geometry*, May, 1993[3].

†estie@ams.sunysb.edu
‡henk@qucis.queensu.ca
§jsbm@ams.sunysb.edu
¶daver@qucis.queensu.ca
‖skiena@cs.sunysb.edu

ABSTRACT

A fundamental problem in model-based computer vision is that of identifying which of a given set of geometric models is present in an image. Considering a "probe" to be an oracle that tells us whether or not a model is present at a given point, we study the problem of computing efficient strategies ("decision trees") for probing an image, with the goal to minimize the number of probes necessary (in the worst case) to determine which single model is present. We show that a $\lceil \lg k \rceil$ height binary decision tree always exists for k polygonal models (in fixed position), provided (1) they are non-degenerate (do not share boundaries) and (2) they share a common point of intersection. Further, we give an efficient algorithm for constructing such decision tress when the models are given as a set of polygons in the plane. We show that constructing a minimum height tree is NP-complete if either of the two assumptions is omitted. We provide an efficient greedy heuristic strategy and show that, in the general case, it yields a decision tree whose height is at most $\lceil \lg k \rceil$ times that of an optimal tree. Finally, we discuss some restricted cases whose special structure allows for improved results.

1. Introduction

In computer vision, one is interested in devising algorithms that will automatically interpret the contents of a digital image (a "scene"). *Model-based* computer vision assumes that we are given some information a priori about the objects for which we are looking in the scene; in particular, we are given a *library* S of k *models* and we are asked to determine which of them are present in the given image. In general, S may be a set of solid models produced by any standard CAD system, and the instances of the models present within the scene may be translated, rotated and scaled copies of elements of S.

In this paper, we examine a fundamental instance of the model-based computer vision problem. Assume that each model in the library S is given in a fixed position, orientation, and scale, and that within the scene there is exactly one instance of one model, and there is no "noise" present in the scene. Our problem is to determine the model, M^*, that is present in the scene by asking a sequence of "probe queries" of the following form: "Is there an object at location p in the scene?" We assume that there is an oracle that answers these probe queries, and we measure complexity in terms of the number of queries to the oracle in order to identify M^*. In practice, such an oracle may be implemented as a local operator on a digitized image — e.g., as a measure of local texture or of gradient field.

A probing strategy is an interactive algorithm that can most naturally be thought of as a binary *decision tree*, in which each node corresponds to a set of candidate models. The root corresponds to the full set S; the leaves of the tree correspond to individual models. Each internal node has an associated probe point that specifies the query that we ask the oracle at that particular stage of the identification. A path from the root to a leaf in the decision tree represents a possible outcome for a particular scene. An example is illustrated in Figure 1.

In this paper, we study the complexity of constructing minimum height decision trees for geometric objects. In other words, given a set S of geometric models, we want to construct (off line) a decision tree so that the worst-case number of probe queries needed to identify M^* is as small as possible.

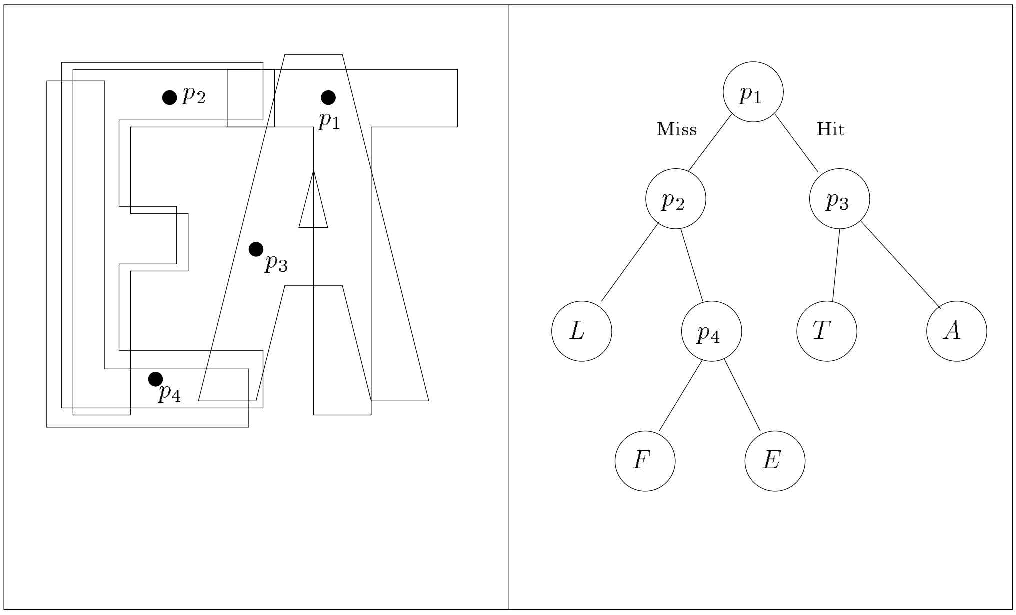

Figure 1: An example of a set S of 5 polygonal models (left) and a probe tree (right) that can determine which model is actually present.

Main results. We formulate and study the decision tree problem from the point of view of *geometry*, proving several related results:

- For a *non-degenerate* set of k polygons in the plane (with no two polygons sharing a common edge) all of which contain the origin, there exists a decision tree of height $\lceil \lg k \rceil$ for S. We show that such an optimal tree can be constructed in time $O(n \lg k)$, where n is the total number of vertices in the polygons S and that there is a lower bound of $\Omega(n + k \lg k)$ on the time to construct a decision tree (of any height).
- We prove that the problem of constructing a minimum height decision tree for a given set of (possibly degenerate) geometric models is NP-complete, even if all models are convex polygons and share a common point. We also prove that if the models are non-degenerate (even axis-parallel unit squares) but share no common point, the problem remains NP-complete.
- We define a very natural "greedy" heuristic for constructing decision trees, and we prove that it yields a decision tree of height at most $\lceil \lg k \rceil$ times that of an optimal decision tree. While the greedy strategy is well-known, to our knowledge, no previous proofs were known that the greedy strategy yields a guaranteed approximation. We also show that there are geometric instances of the problem for which the greedy heuristic attains the worst-case factor $\Theta(\lg k)$.

- We show that in the geometric instance with k simple polygons having a total of n vertices, the greedy heuristic decision tree can be constructed in time $O(hn^2)$, where h is the height of the output tree.

- We give specialized results for various cases of the problem:

  - S is a non-degenerate set of k polygons that are "stabbed" by s points. Then height $s - 2 + \lceil \lg \lfloor k/s \rfloor \rceil$ is sufficient and sometimes necessary for any decision tree for S.

  - S is a set of distinct axis-parallel hyperrectangles in E^d having a common point. Then, a decision tree of height at most $2d\lceil \lg n \rceil$ for S can be constructed in $O(d \cdot n \lg n)$ time.

  - S is a set of n distinct intervals of a line, all sharing a common point. Then, we show that a minimum height decision tree for S can be constructed in $O(n^5)$ time.

  - S is a set of n distinct intervals of a line. Then, we show that a decision tree of height at most 3 times optimal can be constructed in $O(n \lg n)$ time.

Motivation. We are motivated by real instances of the model-based computer vision problem. The specific instance of our problem (with the assumption that the models be given in a fixed position and orientation) arises in recent approaches to model-based vision suggested by Arkin and Mitchell[1], Bienenstock et al.[6,7] (for character recognition), Mirelli[30], and Papadimitriou[32]. For example, Papadimitriou suggests a "probing scheme" in which one searches for instances of geometric models anywhere within an image, using our same model of probing, and he reduces the problem to exactly the problem studied here. In effect, the probing schemes of Ref.1,6,7,32 serve to "factor out" the effect of translation and rotation, reducing the final decision problem to that of this paper. The effect of translation and rotation can also be accommodated within our framework by replicating the model instances according to all possible positions within the image.

Relation with previous work. There is a considerable literature on the subject of classification and regression trees, and their application to statistical decision theory; see Ref.8,12,31,33. Previous work has focused primarily on non-geometric instances of the problem. The *abstract* decision tree problem takes as input a finite *universal* set $X = \{1, \ldots, k\}$ and a family of subsets of X, $\mathcal{T} = \{T_1, T_2, \ldots, T_m\}$, representing the set of possible probes. Hyafil and Rivest[25] prove that it is NP-complete to construct a minimum height or a minimum external path length decision tree. Garey[18] presents dynamic programming algorithms for determining an optimal weighted decision tree. Our problem considered here can be viewed as the unweighted decision tree problem in which the set of possible tests is defined by the faces in the *arrangement* $A(S)$ determined by a set S of k geometric objects.

Recent work on geometric object-identification has been motivated by applications in model-based computer vision[11,23] and tactile sensing in robotics[19,22]. Joseph and Skiena[27] show that $n+3$ finger probes are sufficient and $n-1$ necessary to determine a convex n-gon P selected from a finite set Γ, improving an earlier result by Bernstein[5]. Lyons and Rappaport[29] take S to be a collection of k convex polygons with fixed orientation on a plane, and prove that $k-1$ finger probes are necessary and sufficient to determine the model. An interesting feature of their result is that it is independent of the number of sides in the models; this property is lost when the models may assume arbitrary orientations in the scene. See Skiena[34] for a survey of related results in geometric probing.

Goodman and Smyth[21] give an analysis on the information gain using the greedy algorithm in noisy environments. In this paper, however, we consider the combinatorial problem, using perfect tests.

Belleville and Shermer[4] consider a problem similar to ours: Given a library S of polygons in the plane, identify which one is actually present in a scene. Their goal is to find a set of probe query points, S^*, of minimum cardinality, such that, the answers to the *batch* of queries S^* is sufficient to determine uniquely which polygon is present in the scene. They show that, under two different probing models, this problem is NP-complete. The problem we consider in this paper differs from that of Ref.4 in that we are interested in constructing decision trees that specify *interactive* algorithms for identification, relying heavily on the information returned by previous probe queries in order to select a good choice of the current probe query. In contrast, the method considered in Ref.4 requires that all probes are made simultaneously. As an example to illustrate the difference, consider the set S of one-dimensional models consisting of the k intervals $\{(5i, 5i+2) \ : \ i = 1, \ldots, (k/2)\} \cup \{(5i+1, 5i+3) \ : \ i = 1, \ldots, (k/2)\}$. An optimal batch strategy requires $k-1$ probes. But a decision tree can be constructed of height $k/2+1$, based on probing at the points $5i+1.5$.

An amusing instance of the decision tree problem is the popular game *Battleship*, in which the player is confronted with a 10×10 grid containing five disjoint "ships", each consisting of a horizontal or vertical strip of grid squares (of sizes 5, 4, 3, 3, and 2). The player repeatedly asks whether a particular square on the grid contains a ship, with the goal of identifying as quickly as possible the exact position of each of the ships. Fiat and Shamir[15] study an Israeli version of *Battleship* in which the *area* but not the *shape* of a single rectangular ship is known, and the goal is to minimize the number of probes until the first probe contacts the ship.

2. Arrangements of Polygonal Models

Let S be a set of k simple polygons in the plane, having a total of n vertices. We let $A(S)$ denote the *arrangement* induced by S; $A(S)$ is a collection of 0-faces (vertices), 1-faces (edges), and 2-faces (cells). All points within an edge or a cell of $A(S)$ intersect the same set of polygons of S, and therefore each point in an edge or a cell has the same discriminating power when it is used as a probe point. Thus, the set of possible probes can be identified with the faces (vertices, edges, and cells) of $A(S)$. We define the *cardinality* of a face in $A(S)$ to be the number of elements of S

that contain the face. Note, however, that two different faces may intersect exactly the same subset of S, and will therefore represent equivalent probe possibilities. Also, if cell a intersects $S_a \subseteq S$ and cell b intersects $S_b = S \setminus S_a$, then cells a and b have equivalent discriminating power as probes. For example, in Figure 2, we have labeled cells with their cardinalities; the cell labeled "0" and the two cells labeled "3" all have the same discriminating power as probes.

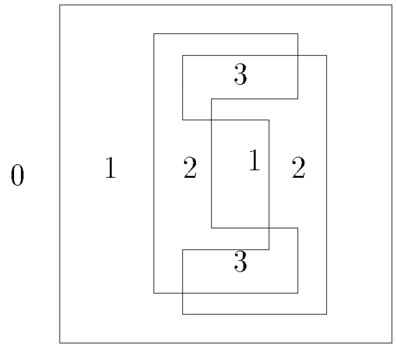

Figure 2: An arrangement of three models: Cells are labeled with their cardinalities.

We say that $A(S)$ (or S) is *degenerate* if two distinct edges of polygons in S intersect in more than a single point (i.e., they intersect in a line segment). We now see that non-degenerate arrangements allow us to construct binary decision trees of the minimum possible height ($\lg k = \log_2 k$) if there is a point in common to the polygons of S.

Theorem 1 *Let S be a non-degenerate set of k polygons in the plane, such that there exists a point p that is in the interior of each polygon. Then there exists a decision tree of height $\lceil \lg k \rceil$ for S.*

Proof. We label each face of $A(S)$ with its cardinality (i.e., the number of polygons that contain it). The cell containing p has label "k", while the cell containing the point at infinity has label "0".

Two cells that share an edge have labels that differ by exactly one, since the arrangement is non-degenerate. This implies that any path from p to the cell at infinity (e.g., a ray out of p) that does not go through a vertex of $A(S)$ must pass through a set of cells whose labels include the full range of cardinalities, $\{0, 1, 2, \ldots, k\}$. Thus, there exists a cell of label $\lceil k/2 \rceil$ in $A(S)$, and we can construct a decision tree such that the probe that is specified at the root node corresponds to such a cell. The effect of probing at a cell of label $\lceil k/2 \rceil$ is to partition S into two subsets: those $\lceil k/2 \rceil$ polygons, S_1, that contain the cell (a "hit" response from the oracle), and those $\lfloor k/2 \rfloor$ polygons, S_0, that do not contain the cell (a "miss" response). Every polygon in S_0 or S_1 contains point p; thus, we can repeat recursively to construct a decision tree, whose height will be $\lceil \lg k \rceil$. $\square$

Remarks.

1. The proof is very general — it holds for any finite set of objects in any number of dimensions, provided non-degeneracy is properly defined for $A(S)$. An object need not be connected, and its connected components may be multiply connected (have "holes").

2. In fact, the proof above goes through for any (possibly degenerate) arrangement $A(S)$ with the property that there exists a path from p to the cell at infinity such that the path does not cross any edge of $A(S)$ that is shared by more than one polygon of S (so that the cardinalities of each pair of cells visited consecutively by the path differ by exactly one). The existence of such a path can be tested in time $O(n^2)$ by constructing the arrangement $A(S)$ and then searching its dual graph.

3. In the worst case, the minimum number of cells visited by a path between a cell labeled k and a cell labeled 0 is $\Theta(n)$. (The lower bound follows from considering a spiral that contains p and a very long horizontal rectangle containing p; the upper bound follows from using the path along the rightward ray out of p.)

Theorem 2 *Let S be a non-degenerate set of k polygons in the plane, such that there exists a point p that is in the interior of each polygon. Then a minimum height decision tree (of height $\lceil \lg k \rceil$) can be constructed in time $O(n \lg k)$, where n is the total number of vertices in the polygons S. Furthermore, there is a lower bound of $\Omega(n + k \lg k)$ on the time to construct a decision tree of any height.*

Proof. First, we argue the upper bound. Let ρ denote the rightward ray out of point p; we can orient the x-axis so that ρ does not pass through a vertex of $A(S)$. We know that ρ begins in a cell of $A(S)$ that has cardinality label k, and that ρ eventually enters the cell at infinity, which has cardinality label 0. By the non-degeneracy assumption, each time that ρ passes through an edge of $A(S)$, the cardinality label goes up or down by exactly 1. Let $f_S(x)$ (for $x \geq 0$) denote the cardinality label function along ρ; i.e., $f_S(x)$ denotes the number of members of S that contain the point that is at distance x to the right of p. Note that $f_S(x)$ is a piecewise-constant function of combinatorial complexity $O(n)$.

We can compute $f_S(x)$ by finding and sorting the intersections between ρ and edges of polygons S (in time $O(n)$), while noting the orientation of each edge crossed (so that we know if the label goes up or down at that point of crossing). For any one polygon of S, the intersections with ρ can be sorted via Jordan sorting[17,24] in time linear in the size of the polygon; thus, we can merge these k sorted lists in time $O(n \lg k)$ to obtain the sorted list of all crossings with ρ, and hence the function $f_S(x)$.

We can store $f_S(x)$ as a sorted list of x-values that correspond to crossing points; with each crossing point, we store a pointer to the edge that is being crossed and we store the cardinality value just after the crossing. We refer to this data structure as the *histogram* for S.

Our algorithm is based on divide-and-conquer. We scan the histogram $f_S(x)$ (in $O(n)$ time) to find a value of x, x_S, for which $f(x_S) = \lceil k/2 \rceil$. During this scan, we can also keep track of which polygons of S are present at the current scan point, so we can know the subset of polygons, S_{hit}, that are present at the point x_S. This value x_S determines an optimal probe point for the root node (S) of our decision tree.

We now scan the histogram $f_S(x)$ one more time, constructing as we go the two new histograms, $f_{S_{hit}}(x)$ and $f_{S_{miss}}(x)$, that correspond to the two subtrees that will hang off the root node in our decision tree. If we then continue to solve the problem recursively, we obtain the following relationship on the time, $t(n,k)$, necessary to compute a minimum height tree for k simple polygons having a total of n vertices:

$$t(n,k) \leq t(n_{hit}, \lceil k/2 \rceil) + t(n_{miss}, \lfloor k/2 \rfloor) + O(n),$$

where $n_{hit} + n_{miss} = n$. The solution to this recursion gives $t(n,k) = O(n \lg k)$.

For the lower bound, we first note that we must at least look at all of the data, so $\Omega(n)$ time is required, for if we fail to examine one vertex of some polygon, then an adversary could potentially select this vertex to be in a position that affects our choice of probe points. A lower bound of $\Omega(k \lg k)$ is obtained from sorting: Given k (unsorted) positive numbers, $x_1, \ldots, x_k$, we define S to be the set of k nested squares $\{(x,y) : -x_i \leq x \leq x_i, -x_i \leq y \leq x_i\}$, $i = 1, \ldots, k$. Consider any decision tree for S, and embed it in the plane so that, for each node, the "hit" subtree is placed left of the "miss" subtree. Then it is not hard to see that a preorder traversal of the tree must visit leaves corresponding to squares in sorted order by x_i. $\Box$

It is critical for the above algorithm that the arrangement $A(S)$ be non-degenerate and that all models share a common point p. We now show that if either of these properties does not hold, then constructing a minimum height tree becomes hard. We first consider degenerate arrangements.

The *abstract* decision tree problem takes as input a finite *universal* set $X = \{1, \ldots, k\}$ and a family of subsets of X, $\mathcal{T} = \{T_1, T_2, \ldots, T_m\}$, representing the set of possible probes. Hyafil and Rivest[25] prove that it is NP-complete to construct a minimum height decision tree for this problem. We now show a relationship between the abstract and the geometric versions of the decision tree problem:

Lemma 1 *For any instance of the abstract decision tree problem, consisting of subsets $\mathcal{T} = \{T_1, \ldots, T_m\}$ of the universal set $X = \{1, \ldots, k\}$, there exists an equivalent geometric instance given by a set S of k convex polygons all sharing a common point.*

Proof. Let M denote a regular m-gon, with sides of unit length and centered at $(0,0)$. Consider the sides of M to be indexed $j = 1, \ldots, m$. For edge j of M, let $v_j \notin M$ be the point "just outside" edge j, at distance $\epsilon > 0$ above the midpoint of edge j, and let Δ_j be the triangle determined by edge j and point v_j. Choose $\epsilon > 0$ to be small enough that $M \cup (\bigcup_j \Delta_j)$ is convex.

Let P_i be the convex polygon

$$P_i = M \cup \left[\bigcup_{\{j \ : \ i \in T_j\}} \Delta_j \right],$$

and let S be the set of k convex polygons, $\{P_1, \ldots, P_k\}$. Each polygon P_i will have at most $2m$ vertices, and the arrangement $A(S)$ has exactly $m+2$ cells: one unbounded cell, the cell M, and the m cells given by triangles Δ_j, $j = 1, \ldots, m$. A probe point in cell M gives no information, since all models in S are present at such a point. Similarly, a probe point in the unbounded cell gives no information. But a probe point in triangle Δ_j hits exactly those models indexed by T_j, and cell Δ_j contains all such probe points. Thus, the geometric instance defined by S encodes the abstract instance specified by X and $\mathcal{T}$. □

An immediate consequence of the above lemma is that the geometric decision tree problem is, in general, hard:

Corollary 1 *The problem of constructing a minimum height decision tree for a given set of (possibly degenerate) geometric models is NP-complete, even if all models are convex polygons and share a common point.*

We comment that a similar construction can be used to show that some collections of k (possibly degenerate) geometric models that are convex polygons and share a common point, may require a probe tree of height $k-1$: Consider M a regular $k-$gon, and triangles Δ_j, as in the proof of Lemma 1. Let P_j be the convex polygon $P_j = M \cup \Delta_j$. Now the only probes that gives informations are the probes at triangles Δ_j, and a "no" answer only eliminates model P_j from the list of possible models present. Therefore as many as $k-1$ probes may be necessary.

We now show that even if S is a collection of *non-degenerate, axis-aligned unit squares*, constructing a minimum height tree is hard if there is no point common to all models of S:

Theorem 3 *Let S be a set of k non-degenerate aligned unit squares. Then the problem of constructing a minimum height decision tree for S is NP-complete.*

Proof. The problem is clearly in NP, since a complete set of candidate probe points can be concisely expressed by taking midpoints of segments joining vertices of polygons of S.

To prove the NP-hardness of determining the existence of a decision tree of height $\leq h$, we modify the reduction used by Fowler, Paterson and Tanimoto[16]. They show that 3-SAT can be reduced to the problem of deciding whether K aligned unit squares suffice to cover a given set of points. Almost the same reduction can be made from 3-SAT to the problem of deciding whether K points suffice to cover a given set of aligned unit squares, where a set of points *covers* (or *stabs*) a given set of squares if each square contains at least one of the points. We simplify the reduction by using as our starting point the Planar 3-SAT problem, thus avoiding the need for having special "cross over" gadgets. We also use ideas of Hyafil and Rivest[25], who show that determining the minimum height of a decision tree is NP-hard for the abstract (non-geometric) case.

The input to 3-SAT is a boolean formula, in conjunctive normal form $C = \{c_1, c_2, \ldots, c_m\}$, where each clause c_j is a subset of three literals from the set of variables $U = \{u_1, u_2, \ldots, u_n\}$, and their negations $\bar{U} = \{\bar{u}_1, \bar{u}_2, \ldots, \bar{u}_n\}$. Define a bipartite graph $G = (V, E)$, where $V = U \cup C$ and E consists of exactly those pairs $\{u, c\}$ such that either u or $\bar{u}$ belong to clause c. If G is planar, the problem of determining whether the formula is satisfiable is called Planar 3-SAT, which was shown to be NP-complete by Lichtenstein[28]. We give a reduction from Planar 3-SAT to determining whether a decision tree of height $\leq h$ suffices to determine which of a set of aligned non-degenerate unit squares is present.

Given a formula of Planar 3-SAT, we draw a corresponding plane graph. For each variable node u, we draw a path (a *wire*, in the notation of Ref.16) surrounding the star consisting of u and the edges incident on it. We think of the wire as being perturbed "just outside" the star. For each of these n wires, we place squares along it, creating a chain of squares such that:

1. The squares are aligned, non-degenerate unit squares.
2. Any pair of successive squares along a wire intersect, but no other squares intersect. In particular, each point is in at most two squares, and squares on different wires do not intersect.
3. The number of squares on each wire is even.

We refer to these squares as *variable squares*. Let the number of squares in the wire of variable u_i be $2K_i$. The minimum number of points necessary to cover all squares in a wire is K_i. There are two such coverings that achieve this minimum. Let one correspond to an assignment of *true* to the variable, and let the other designate *false*.

We also have one unit square for each clause, which we call *clause squares*. A clause square intersects the 3 variable wires corresponding to literals in the clause. The wires are arranged so that the clause squares can be covered "for free" if at least one of the wires of the three variables in it is covered by points which correspond to a truth setting of that literal. Thus, the clause and variable squares can be covered by $K = \sum_i K_i$ points if and only if the formula is satisfiable (as in Ref.16). We complete our construction with two additional squares, called the *disjoint squares*, each of which does not intersect any other square. See Figure 4 for an example of the construction. Finally, set $h = K + 1$.

If the formula is satisfiable, we get a covering of all squares by $K + 2$ points (K for the variable and clause squares, and two additional points for the disjoint squares). It is easy to construct a probe tree of height $h = K + 1$, using all covering points, except one of the points stabbing a disjoint square. The root of the tree is a probe at a point covering a clause. If the probe is a hit, a subtree of height 2 is needed. If the probe is a miss, we probe at another point covering a clause, and so on, until all clause squares are ruled out. Then we can continue our probes, if we have not yet had a hit, to other covering points of the variable squares. If all the probes at variable and clause squares miss, one additional probe to one of the disjoint squares will determine uniquely which is the square present. The height of the tree is given by the path in which each probe is a miss, and is thus $h = K + 1$.

Conversely, we need to show that if a probe tree of height h exists, then the formula is satisfiable. Consider the unique path in the tree from the root to a leaf, in which all probes are a miss. These probe points must cover all but one of the squares, otherwise at the leaf we can not distinguish between two or more squares. We claim that the probe tree can be rearranged, if necessary, so that the one square that is not covered is a disjoint square (i.e., neither a variable nor a clause square). This rearrangement is trivial: replace probe(s) to one of the disjoint squares with probe(s) at a point interior to only the missed square. Clearly, this rearrangement does not increase the height of the tree. We thus have $h = K + 1$ points that cover all variable and clause squares and one of the disjoint squares. But, this is only possible if the clause squares are covered "for free", and the variables squares are covered in either the "true" or "false" coverings that are minimal. Thus we get a truth assignment for the variables that shows the formula is satisfiable.

One technical point remains: We must show that the size of the construction is polynomial; i.e., we do not require too many bits to specify the variable and clause squares. To see why this is true, we appeal to the result of Ref.13, noting that the planar graph for the given instance of Planar 3-SAT can be embedded in an integer grid of size $O(n)$-by-$O(n)$, with nodes placed at integer grid points and edges drawn as straight line segments. Now, in such a grid, the smallest non-zero angle that can be determined by any pair of segments is $\theta_{min} = \Omega(1/n^2)$. Consider a vertex u, whose coordinates are integer, and consider the (axis-aligned) square Σ of side length 2 that is centered on u. Refer to Figure 3. Any two angularly consecutive edges incident on u determine a "wedge region" within Σ. (The wedge is a triangle if the edges do not straddle one of the corners of Σ; otherwise, it is a quadrilateral.) The lower bound on θ_{min} implies that there is an $\Omega(1/n^2)$ lower bound on the length, δ, of the shortest side of the wedge region. This implies that there exists an $\epsilon = \Omega(1/n^2)$ (e.g., $\epsilon = \delta/100$ works) such that the wire can be defined to be the boundary of an $O(\epsilon)$-fattening of the star incident on u, and that, by chosing our variable squares to be of size ϵ, we can be assured that the chain of variable squares that follows the wire will not interfere with other chains of squares. Furthermore, the variable squares can all be aligned to a regular grid whose spacing is $O(1/n^2)$. Thus, we can enlarge the diagram by a factor of $O(n^2)$, obtaining an embedding of our construction with the required unit-sized variable squares, and the whole diagram lives on an $O(n^3)$-by-$O(n^3)$ integer grid. $\square$

Remark. An interesting open question is to determine if the minimum height decision tree problem is NP-complete for a set of *intervals* on a line in non-degenerate position. The above proof relies on the hardness of finding a minimum point cover, a problem which is trivially solvable for intervals on a line. Thus, to prove NP-completeness, we would require a new approach.

The construction in the proof above has the feature that the polygons S cannot be stabbed by fewer than a linear (in k) number of points. Assume now that S is

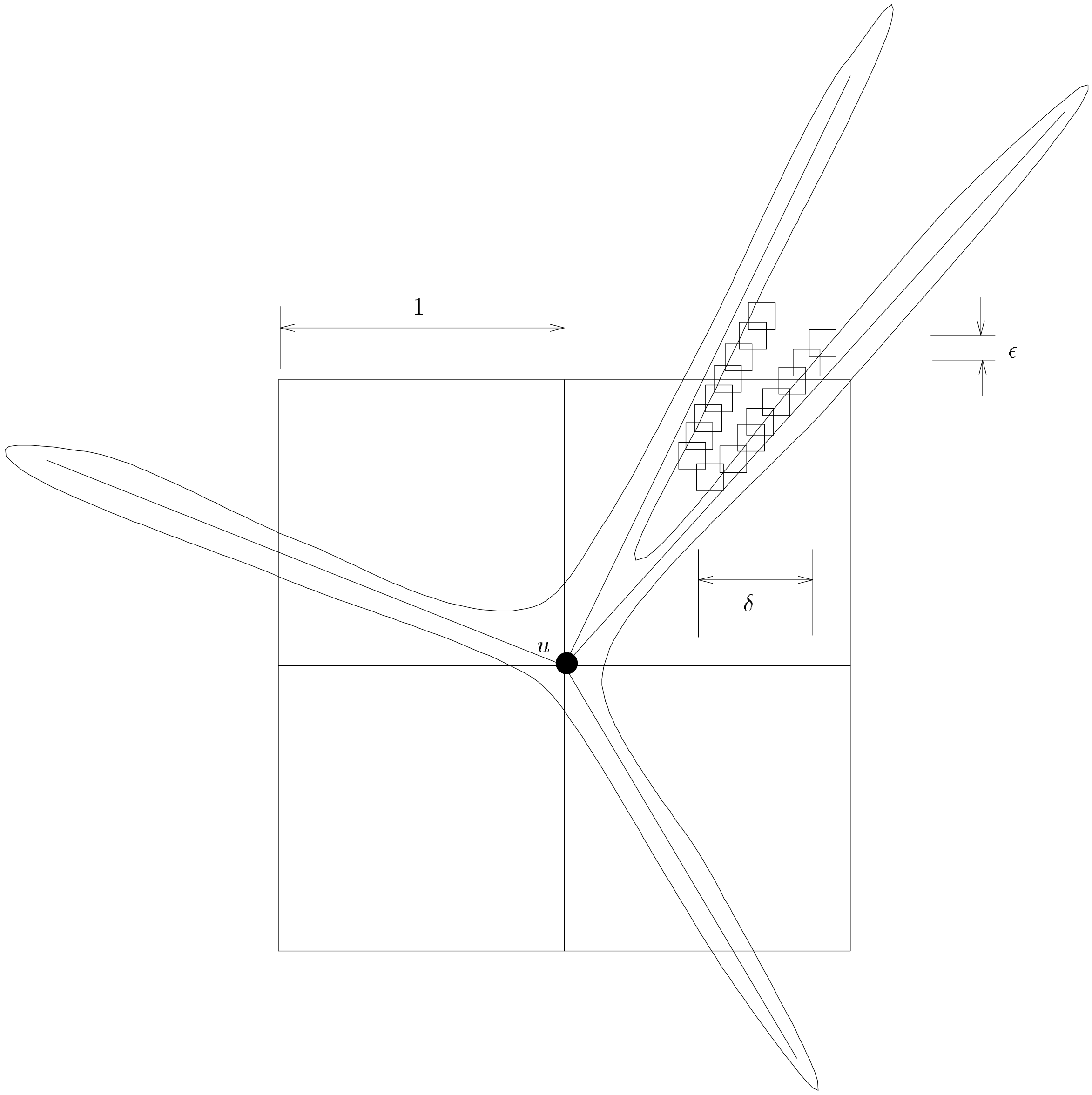

Figure 3: Bounding the size of the grid that contains the construction.

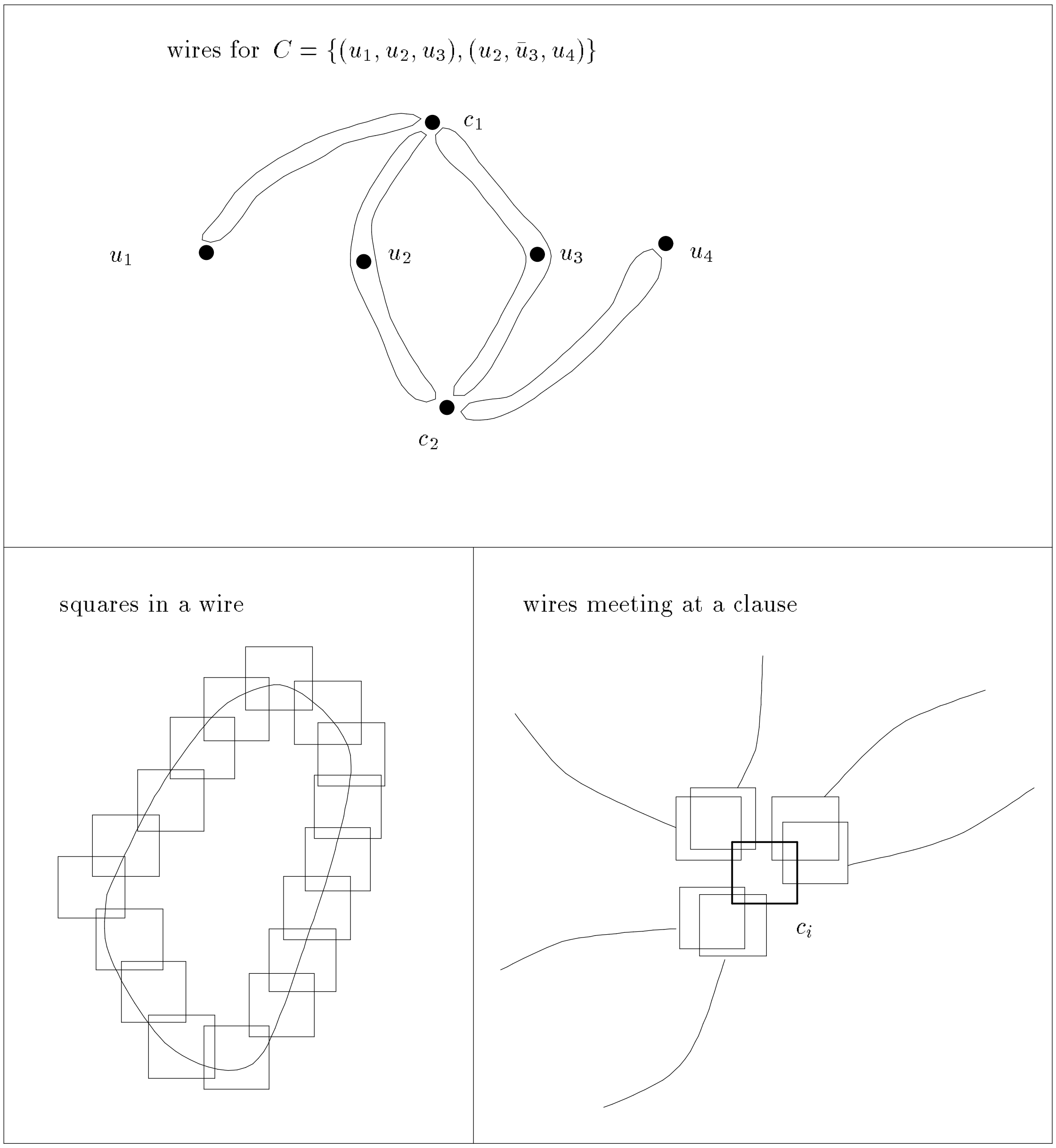

Figure 4: Reduction for non-degenerate aligned unit squares.

in non-degenerate position and is stabbed by a *constant* number, s, of points. The case $s = 1$ is handled by Theorems 1 and 2. If $s = 2$, we claim that the proof of Theorem 1 still holds: One of the two stabbing points must stab at least $\lceil k/2 \rceil$ of the models; if we use this stabbing point as our point p, the argument goes through as before, yielding at most $\lceil \lg k \rceil$ probes.

If $s \geq 3$, then we claim that a decision tree of height $s - 1 + \lceil \lg \lfloor k/(s-1) \rfloor \rceil$ can always be constructed, as follows. Let p_1 be a point that stabs the largest set of polygons in S; let S_1 denote the set of stabbed polygons. Now let p_2 be a point that stabs the largest set of polygons in $S \setminus S_1$; let S_2 denote the set of polygons stabbed by p_2 but not by p_1. Continuing in this way, let p_i be a point stabbing the most polygons in $S \setminus \{S_1 \cup \cdots \cup S_{i-1}\}$ and let S_i be the corresponding set of newly stabbed polygons, for $i = 1, \ldots, s-1$. Note that $|S_1| \geq |S_2| \geq \cdots \geq |S_{s-1}$ and that $|S_i| \leq (n/i)$.

Construct a decision tree based on these choices of probe points p_i. Point p_1 corresponds to the root; the height of the "hit" subtree rooted at p_1 is at most $1 + \lceil \lg |S_1| \rceil$. In general, the length of the longest path from the root of the decision tree to the leaves of the "hit" subtree rooted at p_i is at most $i + \lceil \lg |S_i| \rceil$. Thus, the overall height of the decision tree is at most $(s-1) + \lceil \lg(n/(s-1)) \rceil$. In summary, we have shown:

Theorem 4 *Let S be a non-degenerate set of k polygons in the plane, such that the smallest number of points that stab all members of S is $s \geq 2$. Then height $s - 1 + \lceil \lg \lfloor k/(s-1) \rfloor \rceil$ is sufficient and sometimes necessary for any decision tree for S.*

Remark. Brönniman and Goodrich[9] have recently improved the above bound, for special classes of inputs.

3. Approximating Optimal Decision Trees

Since finding a minimum height decision tree is NP-complete, it is natural to attempt to devise approximation algorithms that are guaranteed to obtain a solution close to optimal. See Moret[31] for a survey of various heuristics for decision tree problems; but no previous methods had proven bounds on their worst-case performance.

We have seen that each candidate probe point partitions the set of objects S into two sets, S_{hit} and S_{miss}. A natural "greedy" heuristic in choosing a good probe point is to do the following: *Select a probe point that partitions the objects as evenly as possible (i.e., minimizes* $\max(|S_{hit}|, |S_{miss}|)$*), and recur.* We then recur on both sides of the probe tree. In this section, we prove that the greedy heuristic always constructs a tree whose height is not more than a small (logarithmic) factor times the optimal height.

For the abstract (non-geometric) version of the problem, with m subsets of a k-element universal set, a greedy tree can be computed in $O(k^2 m)$ time: for each of the $k-1$ internal nodes of the tree, we simply test (in time $O(k)$) each of the m subsets in order to select the best. In our geometric version of the problem, this same approach allows us to construct a greedy tree in time $O(kn^2)$: for each of the $k-1$ internal nodes of the tree, we construct (in time $O(n^2)$) the arrangement of all polygons in the subset of S corresponding to the node, and we search the arrangement for a cell with cardinality closest to half of the subset size. In fact, we can do better than this naive method:

Theorem 5 *For k simple polygons having a total of n vertices, the greedy heuristic decision tree can be constructed in time $O(hn^2)$, where h is the height of the output tree.*

Proof. As in Theorem 2, the idea is to do divide-and-conquer, while maintaining a histogram which gives the number of models at each face of the arrangement. Here, the arrangement, and hence the histogram, are in two dimensions. The histogram of the original arrangement can be constructed in time proportional to the size of the arrangement, $O(n^2)$ (see Ref.10,14). As before, we also maintain with each edge of the arrangement a list of the models whose boundary contains the edge. With this information it is easy to find a face required by the greedy algorithm, and update the histograms, by a simple walk through the arrangement. $\square$

We now show that the greedy heuristic gives a decision tree of nearly optimal height, for either the abstract or geometric version of the problem:

Theorem 6 *For any instance of the decision tree problem on k objects, the greedy heuristic constructs a decision tree of height at most $\lceil \lg k \rceil$ times that of an optimal decision tree.*

Proof. Consider a decision tree constructed by the greedy heuristic. Each node in this tree represents a subset of the set of models, with the leaf nodes of the tree representing singleton sets. For any non-leaf node x, the subsets of its two children form a partition of the subset associated with x. Level l of the decision tree consists of the nodes at a distance of l from the root. Define the *weight* of level l to be the maximum cardinality of any subset on level l of the tree. Let L_i denote the lowest level (smallest l) of weight at most $\lfloor k/2^i \rfloor$.

Let m_i denote the number of tree levels between L_{i+1} and L_i. On the tree level immediately below L_{i+1}, there exists a node c_{i+1} of weight $\lfloor k/2^{i+1} \rfloor + K$, where $K \geq 1$. The path of length $m_i - 1$ from c_{i+1} to its level L_i ancestor defines probes that distinguish at most $\lfloor k/2^{i+1} \rfloor - K$ objects from those in c_{i+1}. Since the greedy heuristic was employed to construct the tree, the number of objects distinguished from c_{i+1} form a non-decreasing sequence along this path. By the pigeonhole principle, the last probe before L_{i+1} distinguishes at most $(\lfloor k/2^{i+1} \rfloor - K)/(m_i - 1)$ models.

We observe that for any instance of I' consisting of all the probes of I but only a subset of its models, the height of the optimal decision tree for I' is no greater than that of I, since adding models cannot make the problem easier. We claim that any tree distinguishing between the $\lfloor k/2^{i+1} \rfloor + K$ models of the ancestor of c_{i+1} must have height at least m_i. Because the greedy strategy was used in constructing the tree, we know that there exists no probe which distinguishes more than $(\lfloor k/2^{i+1} \rfloor - K)/(m_i - 1)$ models from c_{i+1}, so any decision tree for this subset must have height at least

$$\left\lceil \frac{\lfloor k/2^{i+1} \rfloor + K}{(\lfloor k/2^{i+1} \rfloor - K)/(m_i - 1)} \right\rceil \geq m_i$$

Let $h = \max\{m_i : 1 \leq i < \lg k\}$. By the previous discussion, h is a lower bound on the height of the optimal tree. Since h is height of the largest of the $\lceil \lg k \rceil$ portions of the greedy tree, the greedy tree has height at most $\lceil \lg k \rceil \cdot h$. $\square$

To show that the greedy heuristic can lead to a tree with an approximation ratio as bad as $\Omega(\lg k)$, consider the following set of models, which is based on a construction by Johnson[26]. The basic component consists of two rows of $(2^{K+1} - 1)$ unit squares, rotated to resemble diamonds. The squares in each row are arranged in $K + 1$ "groups", such that group i contains 2^i squares, for $0 \leq i \leq K$. The squares within each group are each perturbed slightly, so the arrangement is non-degenerate. All upper squares intersect in a common point, as do all the lower squares. Finally, all squares in group i for both the upper and lower rows have a common point of intersection. Finally, we replicate this arrangement 2^K times, so that each pair of arrangements is non-intersecting. We refer to each of the copies of the arrangement as a "cluster". This construction is illustrated in Figure 5.

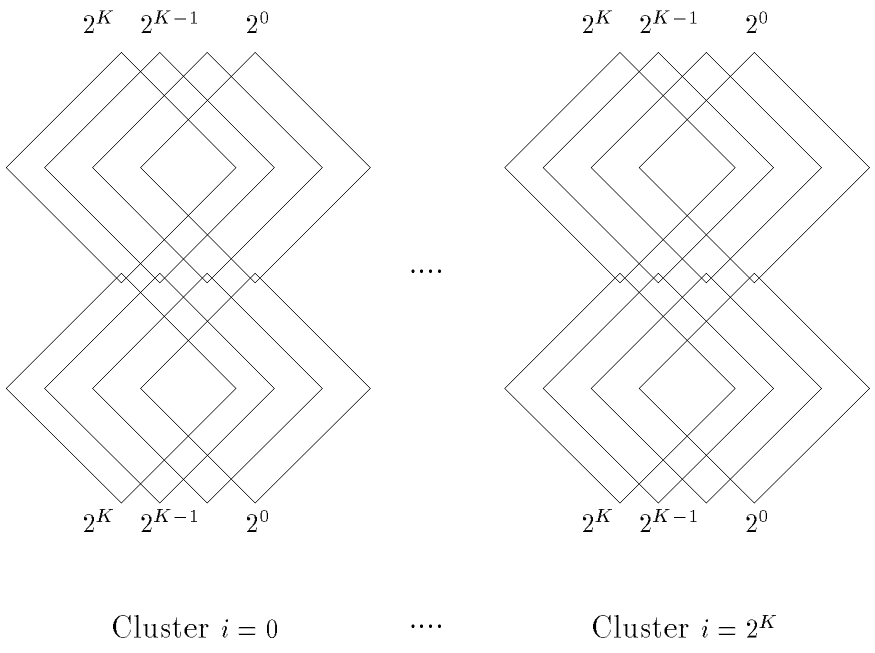

Figure 5: A bad example for the greedy heuristic.

Consider one cluster. The arrangement of squares in this cluster can be covered by two probe points, one at each of the common points of the upper and lower rows, so that each probe hits $2^{K+1} - 1$ squares. A greedy heuristic would instead probe at the common point of the upper and lower group K, which intersects 2^{K+1} squares. Thus a greedy algorithm for point cover would find a cover of cardinality $\sim K/2$ on this example.

Now consider all 2^K clusters. The $k = 2^K \cdot 2(2^{K+1} - 1) \approx 2^{2K+2}$ squares comprise the set of models. The greedy decision tree heuristic will first probe a group K intersection point in each cluster, then the group $K-1$ intersection points, continuing down until the group 1 points. Thus the greedy tree has height $\sim K \cdot 2^K$. However, by first probing at the two cover points for each cluster, and then building a balanced tree for the models in a particular row (which must exist because the arrangement is non-degenerate), we obtain a tree of height $2 \cdot 2^K + \lg(2^{K+1} - 1)$. The ratio of the heights of these trees is $\sim K/2$, where $K \approx (\lg(k) - 2)/2$.

We summarize:

Theorem 7 *There are instances of the geometric decision tree problem for which the greedy heuristic produces a tree whose height is $\Omega(\lg k)$ times that of an optimal tree.*

Remark. We note that this construction can be modified so that every model shares a point in common, without changing the height of either tree. Add a new square to the arrangement, connected to each of the previous squares by a tentacle, such that neither the new square or any tentacle intersect any other square or tentacle. Each model in the new arrangement will consist of an old square, the new square, and the tentacle connecting them. Thus a probe to the new square intersects all the models, a probe to a tentacle intersects exactly one model, and probes to all other regions intersect the same number of models as before. The new options do not permit a more efficient probing strategy.

4. Special Classes of Degenerate Arrangements

As we have seen, a set of k convex polygons sharing a common point of intersection can still require a probe tree of height $k - 1$ if degeneracies occur, whereas a $\lceil \lg k \rceil$ height tree always suffices if they do not. In this section, we consider some other special classes of models for which we give polynomial-time algorithms for the exact or approximate solution of the decision tree problem.

Theorem 8 *Let S be a set of k distinct axis-parallel hyperrectangles in E^d, where each rectangle of S contains a common point p. A decision tree of height at most $2d\lceil \lg k \rceil$ for S can be constructed in $O(d \cdot k \lg k)$ time.*

Proof. Identify the points of intersection between S and a ray from p in each of the isothetic directions. An optimal decision tree on the distinct points of intersection in each direction will have height at most $\leq \lceil \lg k \rceil$, and determine one edge of the model rectangle. Therefore, the union of these $2d$ decision trees determines the model and has the prescribed height. The time of construction is dominated by the cost of sorting the edges of the rectangles in each dimension. □

The simplest class of geometric objects are a set S of k intervals on the line. It is easily seen that the height of the decision tree partially depends upon the number of disconnected regions formed by the union of the intervals of S. If the k intervals of S are mutually disjoint, no decision tree of height less than $k-1$ will distinguish between the models. When the intervals all share a common point of intersection, a minimum height decision tree can be efficiently constructed.

Theorem 9 *Let S be a set of k distinct intervals of a line, all sharing a common point p. A minimum height decision tree for S can be constructed in $O(k^5)$ time.*

Proof. We use dynamic programming. Observe that the arrangement of intervals defines $i+1 \leq k+1$ subintervals to the left of p (labeled l_0 to l_i), and at most $j+1 \leq k+1$ subintervals to the right of p (labeled r_0 to r_j), which together define the possible probe points. For convenience, l_0 and r_j denote the subintervals which do not intersect S, while l_i and r_0 both denote the face containing p. At any point during the execution of a probing strategy on S, the complete state of our knowledge about the query segment is represented by four parameters. Let l_m (r_m) be the rightmost (leftmost) probe to the left (right) of p which does not intersect the query segment, and let l_h (r_h) be the leftmost (rightmost) probe to the left (right) of p which hits the query segment.

Let $T(S)$ be the height of a minimum height tree for S. Then we have $T(S) = T(l_0, l_i, r_0, r_j)$, where

$$\begin{aligned} T(l_m, l_h, r_h, r_m) = \quad \min(& \min_{l_m < l < l_h} (\max(T(l, l_h, r_h, r_m), T(l_m, l, r_h, r_m)) + 1), \\ & \min_{r_h < r < r_m} (\max(T(l_m, l_h, r, r_m), T(l_m, l_h, r_h, r)) + 1)), \end{aligned}$$

and $T(l, l+1, r, r+1) = 0$ for $0 \leq l < i$ and $0 \leq r < j$. This recurrence can be evaluated in time $O(k^5)$ by memoization, or explicitly by evaluating it in terms of increasing $\Delta_l = l_h - l_m$ and $\Delta_r = r_m - r_h$. □

Let us now consider the case in which the intervals no longer share a common point. For a set of intervals S, a *point cover* of S is a set of points, $\{p_1, \ldots, p_s\}$, such that each interval in S contains at least one point p_i (i.e., each interval is "stabbed" by a point p_i). A *minimum point cover* is a point cover of minimum cardinality s. Computing a minimum point cover is equivalent to the problem of minimum clique cover in interval graphs (see Golumbic[20]), and it can be done very efficiently:

Lemma 2 *A minimum point cover for a set S of k intervals on a line can be computed in time $O(k \lg k)$.*

Proof. We sort the interval endpoints, and we scan the sorted list in left-to-right order. When a left endpoint of interval s is encountered, then s is pushed onto a stack. When a right endpoint is encountered, we place a point p_i at this right endpoint and remove from S all of those intervals S_i that are stabbed by p_i. We repeat this until S is empty. It is easily verified that each interval contains at least one point p_i. The optimality of the cover follows inductively from the fact that any cover must include at least one point to the left of the leftmost right endpoint. □

Theorem 10 *Let S be a set of k distinct intervals on a line and let $s \geq 2$ be the size of a smallest point cover of S. A decision tree of height at most $(s-1) + 2\lceil \lg(k/(s-1)) \rceil$ for S can be constructed in $O(k \lg k)$ time, and this tree is at most three times the height of an optimal tree.*

Proof. Let p_1 be a point that stabs the largest set of intervals in S; let S_1 denote the set of stabbed intervals. Now let p_2 be a point that stabs the largest set of intervals in $S \setminus S_1$; let S_2 denote the set of intervals stabbed by p_2 but not by p_1. Continuing in this way, let p_i be a point stabbing the most intervals in $S \setminus \{S_1 \cup \cdots \cup S_{i-1}\}$ and let S_i be the corresponding set of newly stabbed intervals, for $i = 1, \ldots, s-1$. Note that $|S_1| \geq |S_2| \geq \cdots \geq |S_{s-1}|$ and that $|S_i| \leq (k/i)$.

Construct a decision tree based on these choices of probe points p_i. Point p_1 corresponds to the root; the height of the "hit" subtree rooted at p_1 is at most $1 + 2\lceil \lg |S_1| \rceil$, by Theorem 8 (with $d = 1$). In general, the length of the longest path from the root of the decision tree to the leaves of the "hit" subtree rooted at p_i is at most $i + 2\lceil \lg |S_i| \rceil$. Thus, the overall height of the decision tree is at most $(s-1) + 2\lceil \lg(k/(s-1)) \rceil$.

To prove the approximation ratio, we note that $\max(s-1, \lg k)$ is a lower bound on the height of the optimal decision tree for S. It is clear that $3 \cdot \max(s-1, \lg k) \geq (s-1) + 2\lg(k/(s-1))$. $\square$

The decision tree produced by this algorithm is not necessarily optimal, even when the endpoints of the intervals are distinct. The difficulty hinges on the fact that a different decomposition into subsets may result in a shallower decision tree, and the order in which we process the decompositions of S can also improve the height of the decision tree.

5. Conclusions

Our goal in this paper has been to examine the decision tree problem on a set of *geometric* objects. We have seen that, as with abstract decision tree problems, geometric problems are, in general, NP-complete. However, there *is* structure in geometry, as we have seen in several special instances. In particular, we have given an algorithm to construct an optimal $\lceil \lg k \rceil$-height binary decision tree for any set of k geometric models in non-degenerate configuration that are stabbed by a common point (or by two points). When degeneracies occur or when there is no common stabbing point, finding a minimum height tree is NP-complete, so we have given a heuristic that constructs a decision tree whose height is at most $\lceil \lg k \rceil$ times that of an optimal tree. Our algorithms are practical and should be easy to implement.

We close with some open problems:

- Does there exist a polynomial-time algorithm for constructing an optimal decision tree for a (possibly degenerate) set of rectangles sharing a common point?
- Does there exist a polynomial-time algorithm for constructing an optimal decision tree for a set of intervals on a line, *not* necessarily sharing a common point?
- We have shown that non-degenerate arrangements of k objects admit decision trees of height $\lg k$. For what other classes of objects must there always exist $O(\lg k)$ height decision trees?

- Any measurement is potentially subject to noise. Probing in a small face near the boundary of many models will not be as robust as probing in the center of a large interior face. How can "robust", yet efficient, decision trees be constructed?

Finally, we remark that in the time since this paper was first written, an extension has been made to the case of "concept classes" (in which the goal is to identify the class to which an object belongs, rather than to determine the exact identity of an object); see Arkin et al.[2].

6. Acknowledgements

These results grew out of discussions that began in a series of workshops on Geometric Probing in Computer Vision, sponsored by the Center for Night Vision and Electro-Optics, Fort Belvoir, Virginia, and monitored by the U.S. Army Research Office. In particular, we thank Teresa Kipp and Vince Mirelli of the Center for Night Vision and Electro-Optics, and we thank all of the participants of the series of workshops.

The views, opinions, and/or findings contained in this report are those of the authors and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other documentation.

E. Arkin was partially supported by NSF Grants ECSE-8857642 and CCR-9204585. H. Meijer was partially supported by NSERC of Canada Grant 0282. J. Mitchell was partially supported by grants from Hughes Research Laboratories, Boeing Computer Services, Air Force Office of Scientific Research contract AFOSR-91-0328, and by NSF Grants ECSE-8857642 and CCR-9204585. D. Rappaport was partially supported by NSERC of Canada Grant A9204. S. Skiena was partially supported by NSF grant CCR-9109289 and New York Science and Technology Foundation grants RDG-90171 and RDG-90172.

References

1. E. M. Arkin and J. S. B. Mitchell. "Applications of combinatorics and computational geometry to pattern recognition," Technical Report, Cornell University, 1990.
2. E. M. Arkin, M. T. Goodrich, J. S. B. Mitchell, D. Mount, C. D. Piatko, and S. S. Skiena. "Point probe decision trees for geometric concept classes," In *Proc. 3rd Workshop Algorithms Data Struct.*, volume 709 of *Lecture Notes in Computer Science*, pages 95–106. Springer-Verlag, 1993.
3. E. M. Arkin, H. Meijer, J.S.B. Mitchell, D. Rappaport, and S.S. Skiena, "Decision trees for geometric models," In *Proc. 9th Annu. ACM Sympos. Comput. Geom.*, May, 1993, pp. 368–378.
4. P. Belleville and T. C. Shermer. "Probing polygons minimially is hard," *Comput. Geom. Theory Appl.* **2** (1993), 255–265.
5. H. Bernstein. "Determining the shape of a convex n-sided polygon by using $2n + k$ tactile probes," *Inform. Process. Lett.*, **22** (1986), 255–260.
6. E. Bienenstock, D. Geman, and S. Geman. "A relational approach in object recognition," Technical Report, Brown University, 1988.

7. E. Bienenstock, D. Geman, S. Geman, and D.E. McClure. "Phase II technical report, development of laser radar ATR algorithms," Technical Report Contract No. DAAL02-89-C-0081, CECOM Center for Night Vision and Electro-Optics, October 1990.

8. L. Breiman, J.H. Friedman, R. A. Olshen, and C.J. Stone. *Classification and Regression Trees*, Wadsworth Inc., Belmont, California, 1984.

9. H. Brönnimann and M. T. Goodrich. "Almost optimal set covers in finite VC-dimension," *Discrete Comput. Geom.*, **14** (1995), 263–279.

10. B. Chazelle and H. Edelsbrunner. "An optimal algorithm for intersecting line segments in the plane," *J. ACM*, **39** (1992), 1–54.

11. R. Chin and C. Dyer. "Model-based recognition in robot vision," *ACM Comput. Surv.*, **18** (1986), 67–108.

12. P. A. Chou. "Optimal partitioning for classification and regression trees," *IEEE Trans. Pattern Anal. Mach. Intell.* **13** (1991), 340–354.

13. H. De Fraysseix, J. Pach, and R. Pollack. "How to draw a planar graph on a grid," *Combinatorica*, **10** (1990), 41–51.

14. H. Edelsbrunner. *Algorithms in Combinatorial Geometry*, Springer-Verlag, Heidelberg, 1987.

15. A. Fiat and A. Shamir. "How to find a battleship," *Networks*, **19** (1989), 361–371.

16. R. J. Fowler, M. S. Paterson, and L. Tanimoto. "Optimal packing and covering in the plane are NP-complete," *Inform. Process. Lett.*, **12** (1981), 133–137.

17. K. Y. Fung, T. M. Nicholl, R. E. Tarjan, and C. J. Van Wyk. "Simplified linear-time Jordan sorting and polygon clipping," *Inform. Process. Lett.*, **35** (1990), 85–92.

18. M. Garey. "Optimal binary identification procedures," *SIAM J. Appl. Math.*, **23** (1972), 173–186.

19. P. Gaston and T. Lozano-Perez. "Tactile recognition and localization using object models: The case of polyhedra on a plane," *IEEE Trans. Pattern Anal. Mach. Intell.*, **6** (1984), 257–266.

20. M. C. Golumbic. *Algorithmic Graph Theory and Perfect Graphs*, Academic Press, New York, NY, 1980.

21. R. Goodman and P. Smyth. "Decision tree design from a communication theory standpoint," *IEEE Trans. Inform. Theory*, **34** (1988), 979–994.

22. W. E. Grimson. "The combinatorics of local constraints in model-based recognition and localization from sparse data," *J. ACM*, **33** (1986), 658–686.

23. W. E. Grimson and T. Lozano-Perez. "Localizing overlapping parts by searching the interpretation tree," *IEEE Trans. Pattern Anal. Mach. Intell.*, **9** (1987), 469–482.

24. K. Hoffmann, K. Mehlhorn, P. Rosenstiehl, and R. E. Tarjan. "Sorting Jordan sequences in linear time using level-linked search trees," *Inform. Control*, **68** (1986), 170–184.

25. L. Hyafil and R. Rivest. "Constructing optimal binary decision trees is NP-complete," *Inform. Process. Lett.*, **5** (1976), 15–17.

26. D. S. Johnson. "Approximation algorithms for combinatorial problems," *J. Comp. Sys. Sciences*, **9** (1974), 256–278.

27. E. Joseph and S. Skiena. "Model-based probing strategies for convex polygons," *Comput. Geom. Theory Appl.*, **2** (1992), 209–221.

28. D. Lichtenstein. "Planar formulae and their uses," *SIAM J. Comput.*, **11** (1982), 329–343.

29. K. Lyons and D. Rappaport. "An efficient algorithm for identifying objects using robot probes," *Visual Comput.*, **10** (1994), 452–458.

30. V. Mirelli. "Computer vision is a highly structured optimization problem," Manuscript, Center for Night Vision and Electro-Optics, Fort Belvoir, VA, 1990.

31. B. Moret. "Decision trees and diagrams," *ACM Comput. Surv.*, **14** (1982), 593–623.

32. C. H. Papadimitriou. "On certain problems in algorithmic vision," Technical Report, Computer Science, Univ. California San Diego, 1991.

33. S. R. Safavian and D. Landgrebe. "A survey of decision tree classifier methodology," *IEEE Trans. Syst. Man Cybern.*, **21** (1991), 660–674.

34. S. S. Skiena. "Interactive reconstruction via probing," *IEEE Proceedings*, **80** (1992), 1364–1383.