Geometric Optimization And Computational Complexity

Chanderjit Bajaj
Ph.D. Thesis

TR 84-629
August 1984

Department of Computer Science
Cornell University
Ithaca, New York 14853
An appeal, 1881

Chancellor Hopl

by

Doctor of Philosophy

in partial fulfillment of the requirements for the degree of

Doctoral University

Presented to the Faculty of the Graduate School

A Thesis

Comparative Complexity

and

Comparative Optimization
Biographical Sketch

Chanderjit Bajaj was born in Calcutta, India on April 19, 1958.

His junior years were spent in St. Xavier's School, Calcutta. After graduating from St. Columba's High School in New Delhi, he attended the Indian Institute of Technology, New Delhi where he graduated in 1980 with a B-Tech. (Bachelor of Technology) degree in Electrical Engineering. Subsequently, he attended the University of Pittsburgh and received an M.S. in Computer Science in 1981.

Bajaj began his graduate studies in Computer Science at Cornell University in 1981. He received an M.S. in Computer Science in 1983.

Dedication

To the fond memory of my father, Shri Madan Lal Bajaj, who was also my dearest friend.
Acknowledgements

I would like to thank John Hopcroft for his patience, teaching and encouragement throughout my term at Cornell. He taught me how to approach seemingly difficult problems and particularly to 'winnow the chaff from the grain', in my research. For his advice and constant suggestions, on and off the field of research, I shall always be grateful. David Gries befriended me, as he does most all incoming students, from my first day at Cornell. From him I not only learnt how to play softball (third base), but also the science of teaching (and programming). As a member of my committee, he also promises to read my thesis. So does Chris Pottle, who I also wish to thank for cheerfully serving on my committee.

Paul Pritchard, (or should I say Geoff Hunt ?), the Squash champ and Juris Hartmanis, the Tennis champ, I wish to thank for the numerous discussions before, after and at times during our championship duels. Ming Li (the best Chinese cook in town) and Luc Longpre (the best French cook around), I thank for the wonderful animated conversations we had, ranging from the highly technical to the deeply philosophical and murderously mundane. Our frequent dinners together were highly conducive to research and I shall miss them. Tom Coleman, John Gilbert, Raimund Seidel and Yacov Yesha are the other persons who were always ready to listen to an interesting thought and also provided me with many a stimulus.

To my father my revered thanks, for all that you taught me and all the great times we shared together. For the rest of my family consisting of my mother, three sisters, (and four nephews), I have the deepest love and sincere thanks for being my staunchest friends. For Khawro Rustamji, who has been a second father to me, I have only the utmost admiration. For your unselfish help and guidance, thanks don't seem enough.

My other friends (and family in Ithaca) that I wish to thank, consist of Paul, Cathy and Julie Dawson, Megu (haha), Big Cy, small Cy, Punj, Daf, Guy, Alka, Reiko (fair weather) Yamada and Elena Pasquale (Are you going to return to Italy or is it La Jolla?).

It would be remiss to exclude all the nice compadre's of our Cornell Computer Science department. For fear of being clubbed over the head on leaving someone's name out, let me offer my warmest thanks collectively to all bridge groups, volleyball cliques, tennis factions, softball squads, badminton trios, basketball slammie-jammers, etc., that this wonderful place terms with. Finally sincere thanks to Mike Woodall, our proficient mathematics librarian, for all the help he provided and who never protested even my most obscure demands.
Table of Contents

1. INTRODUCTION ....................................................................................... 1

1.1. Geometric Optimization ................................................................. 1

1.2. Location-Allocation Problems ..................................................... 5

2. GEOMETRIC OPTIMIZATION AND THE POLYNOMIAL HIERARCHY .... 8

2.1. Overview ...................................................................................... 8

2.2. MINSUM Geometric Optimization Problems ................................. 9

2.2.1. Decision Problems ................................................................. 11

2.2.2. Strong Completeness .............................................................. 23

2.3. MINMAX Geometric Optimization Problems ............................... 28

2.3.1. The Euclidean distance metric .............................................. 31

2.3.2. The \(l_p, 1 \leq p \leq \infty\) distance metrics ............................... 41

2.3.3. Arbitrary distance metrics .................................................... 42

2.3.4. Appendix A ............................................................................ 45

3. EFFICIENT ALGORITHMS AND LOWER BOUNDS ..................... 47

3.1. Background ................................................................................ 47

3.2. Geometric Location .................................................................... 49

3.2.1. Locating under MINMAX .................................................... 51

3.2.2. Locating under MINSUM ..................................................... 57

3.3. Related Geometric Optimization Problems .................................. 61

4. ALGEBRAIC COMPLEXITY OF GEOMETRIC OPTIMIZATION .......... 64

4.1. Introduction .................................................................................. 64

4.2. The Fermat-Weber problem ....................................................... 65

4.3. The Line-restricted Weber problem ........................................... 78

5. REFERENCES .................................................................................... 83
In the field of geometric optimization problems, the complexity and methods used to solve these problems are of great importance. The focus on geometric optimization problems has led to the development of new algorithms and techniques that can efficiently solve these problems. One of the key problems in geometric optimization is to find the minimum or maximum of a function defined on a geometric object. This often involves finding the shortest path or the largest area within a given set of constraints.

For instance, in the context of geometric optimization, finding the shortest path between two points on a surface or within a polyhedron is a fundamental problem. This can be approached using various algorithms, such as the Dijkstra's algorithm for shortest paths or the convex hull algorithm for finding the boundary of a set.

Moreover, the complexity of these problems often depends on the specific constraints and the geometric properties of the objects involved. The study of these problems not only has theoretical significance but also finds applications in various fields, including computer graphics, robotics, and engineering design.

In summary, the field of geometric optimization is a rich area of research with a wide range of applications. The challenges posed by these problems continue to drive the development of new algorithms and methodologies, making it an active and exciting field of study.
CHAPTER 1

INTRODUCTION

1.1. Geometric Optimization

Geometry is perhaps the oldest branch of mathematics that has not lost its lustre through the period of time. It is the study of the size, position, orientation and shape of most of the familiar objects around us. Geometrical figures like circles, ellipses, squares, rectangles are easily discernible in most of our modern day amenities. The wheels of a car were chosen to be roughly circular since circles roll easier than irregularly shaped objects. Man, being an optimizer (a maximizer or minimizer), always optimizes the geometric objects around him, through a search for regularity and symmetry and to make his task easier. Typical geometric optimization problems are the problem of packing the maximum number of oranges (spheres) into a rectangular crate and that of paving an oval shaped floor using the minimum number of square tiles. Minimizing operation costs is often also an optimization objective.

During the last few years sophisticated computer applications involving the representation and manipulation of geometric objects, have rapidly appeared in a broad variety of disciplines. Areas such as robotics, CAD/CAM, computer graphics and office automation are some of the major forerunners. For all of them, geometric optimization is a common theme. In robotics a long desired goal is that of having a robot pick up workpieces from individual bins, contour them to the right geometric shape (for example a bracket with bevelled cylindrical holes and spiral screw threads), with the minimum amount of waste material and the maximum degree of accuracy, and finally assemble them into the finished product. In CAD/CAM, designers give shape to automobiles with the help of wide variety of geometric figures consisting of planar and quadric surfaces, while minimizing the drag coefficient and maximizing the esthetic appeal. Computer graphics and office automation are involved with problems of displaying numerous geometric figures with maximum accuracy and minimum cost and time. In order that these areas achieve their full promise, it will be necessary to design efficient algorithms for the problems that arise as well as classify the inherent computational complexity of the more difficult ones.

Geometric Location-Allocation Optimization Problems involve the optimization of the placement of various geometric objects and are of considerably renewed importance, precisely for the above reasons. They are of practical importance because such problems are frequently formulated in the location of facilities, such as water treatment plants, oil refineries, industrial warehouses, fire stations and hospital clinics, [Co63],[FW74],[Ha63],[KP79],[Sc71],[We57]. More recent applications arise from VLSI layout design (component location) and robotics (robot location) [GW80],[IA89]. The strong theoretical interest stems from the fact that though these optimization problems have occurred and been studied for quite some time in a diversity of literature in network location theory, [FW74],[KP79],[TP80], little is known about their computational complexity, in particular for the important case of the Euclidean plane. Furthermore, since nearly all of these problems are recast in a mathematical framework bordering on Euclidean geometry these location problems, are mathematically intriguing in their own right.
and geometric optimization problems, which are closely related to these problems. We also
considered some geometric optimization problems that are closely related to the
geometric optimization problems considered in this paper.

In conclusion, we have shown that the geometric optimization problems considered in
this paper are closely related to geometric optimization problems that are closely related to
the geometric optimization problems considered in this paper.
more, necessary conditions for the existence of maxima and minima in optimization problems are generally tied to the question of solvability of an equation or a system of equations. In calculus these equations are algebraic; (in the calculus of variations, they are differential). By generating the minimal polynomial, whose root over the field of rational numbers is the solution of the geometric optimization problem on the real (Euclidean) plane, we are able to answer questions about the solvability (or nonsolvability) of the problem by radicals. Furthermore, the algebraic degree of the optimizing solution, which is the degree of the irreducible minimal polynomial for the problem, and the order of the splitting field of the minimal polynomial, which is also the order of the appropriate Galois group, gives an idea of the inherent difficulty of constructing the solution as well as gives a complexity measure for these geometric optimization problems.

In light of the above, a more concerted application of algebraic field extensions (and Galois Theory) to geometric optimization problems seems to be a very worthwhile objective. A step in this direction is undertaken in [Chapter 4], wherein we reduce certain geometric optimization problems to algebraic problems and analyze their complexity as outlined above.

1.2. Location-Allocation Problems

There frequently arise in practice problems that are concerned with how to serve and supply, in an optimum fashion, a set of destinations that have fixed and known locations. What must be determined, in these problems, is the number, location and size of the sources that will most economically supply the given set of destinations with some commodity, material or service. Examples of such problems are the determination and location of warehouses, distribution centers, communication centers and production facilities.

Location - Allocation problems are characterized by the following general structure.

Given a set of n points (destinations) in the plane, find the location of k points (sources) and an allocation of each destination to a source, so as to minimize an objective function based on the distances between sources and destinations.

It should be noted that in many cases the full problem in all its generality seldom arises. The assumptions made for these problems are as follows:

1. There are no restrictions on the permissible source capacities.
2. Unit shipping costs are independent of total source output.

Even with these assumptions the problem is a formidable one.

If it is assumed that there are no capacity restrictions and that unit shipping costs are independent of source output, it is clear that in a minimum cost solution, each destination will be supplied by a single source, moreover, by its closest source. However, in addition to not knowing the location of each of the m sources in the minimum cost solution, we also do not know which source is to serve which subset of destinations. In order to avoid a definite association of sources and destinations at the outset, we shall assume in the definition of the objective function (cost of supplying n destinations with m sources) that every destination can be supplied by any source.

For k sources and n destinations there are S(n,k) possible assignments of n destinations to k sources, where S(n,k) is the Stirling number of the second kind and is given by

\[ S(n,k) = \frac{1}{k!} \sum_{j=0}^{k} (-1)^j \binom{k}{j} j^k \]
CHAPTER 2

GEOMETRIC OPTIMIZATION AND THE POLYNOMIAL HIERARCHY
In section (2.3) one again shows that if \( NP \neq Co \ NP \) then there are interesting natural problems properly in \( \Delta^p_2 \) and furthermore they are complete for a class \( D^p \) (which is contained in \( \Delta^p_2 \) and contains \( NP \cup Co \ NP \)). Three different classes of geometric optimization problems derived under a discrete minmax optimization criterion from location-allocation problems in the Euclidean plane, are considered. We call these problems, 'MINMAX Geometric Optimization Problems'. These problems are shown to be polynomial-time reducible to each other and furthermore \( D^p \) complete. In particular for \( l^p \) distances, \( 1 \leq p \leq \infty \), and arbitrary distance metrics as well as for problems having either finite or infinite feasible solution sets.

2.2. MINSUM Geometric Optimization Problems

A minsum location objective is one which minimizes the sum of the costs resulting from a given location solution and is some measure of the average cost of serving the destinations. A common real world situation that might be formulated with a minsum objective is one of locating a water treatment plant so that the sum of the length of pipes required to serve water to the various households or industrial users is minimized. Another example could be that of locating components on a VLSI chip so as to minimize the sum of the wire connections from the components.

Under this minsum location objective it is possible to distinguish two basic approaches. The first suggests that location sites may be anywhere in the real Euclidean plane, giving an infinite number of possible location sites. The second approach considers only a finite number of known sites as feasible and models the constraints imposed on the possible location, ensuring that undesirable and impractical locations need not be considered. The various distance metrics used, Rectilinear \( (l_1) \), Euclidean \( (l_2) \) and \( \infty \) \( (l_\infty) \), reflect the appropriate problem restrictions. Between two points \( a = (x, y) \) and \( b = (b_x, b_y) \) in the plane the \( l_1 \) distance is \( |x - b_x| + |y - b_y| \); the \( l_2 \) distance is \( \sqrt{(x - b_x)^2 + (y - b_y)^2} \) and the \( l_\infty \) distance is \( \max(|x - b_x|, |y - b_y|) \).

Here we consider the three different classes of multiple points location-allocation problems, \( (P_1, P_2 \) and \( P_3) \) in the plane under a minsum optimization objective. In these problems we are given a set \( T = \{(x_i, y_i), i = 1..n\} \), the location of \( n \) fixed destination points \( (destinations) \) in the plane and need to locate a number of points \( (sources) \) to service the destinations in a way which behooves the application on hand. In the case of locating multiple sources, the allocation of the destinations to the sources must also be ascertained. A common assumption for these problems [Co63] is that the sources are considered to be uncapped; that is there are no capacity limitations. As a consequence, each destination can be completely serviced by a single source though a source can itself service more than one destination. Furthermore in the optimal solution each destination is allocated to its closest located source. However this optimal allocation, is one of the exceedingly large number of possible allocations\(^1\). Not known a priori it needs to be determined. It is also interesting to note that the capacitated versions of these geometric location-allocation optimization problems (with sources having finite capacities), turn out to be various cases of the more familiar transportation location problems and under discrete solution space constraints, to be the plant location and warehouse location problems [FW74].

\(^1\) The total number of possible assignments (allocations) of \( n \) destinations to \( k \) sources is \( S(n, k) \), the Stirling number of the 2nd kind.
Section 2.3.1. Decision Problems

Consider the problem of finding the best decision in a decision-making process. The solution to this problem is given by the decision function $f$. The decision function is defined as

$$ f(x) = \text{argmax}_{y} P(Y = y | X = x) $$

where $P(Y = y | X = x)$ is the probability of outcome $y$ given input $x$. The decision function $f$ is used to choose the best action $y$ given the current state $x$.

The problem of finding the best decision can be formulated as an optimization problem:

$$ \max_{y} P(Y = y | X = x) $$

subject to the constraint $y \in \mathcal{Y}$, where $\mathcal{Y}$ is the set of all possible outcomes.

The solution to this optimization problem is given by the decision function $f(x)$. The decision function $f$ is used to choose the best action $y$ given the current state $x$.
(2) \( P \) is not in \( \Sigma_1^P = \mathsf{NP} \), assuming \( \mathsf{NP} \not\subset \mathsf{Co-NP} \).

(3) \( P \) is not in \( \Pi_1^P = \mathsf{Co-NP} \), assuming \( \mathsf{NP} \not\subset \mathsf{Co-NP} \).

To show the above for our optimization problems we first need to show the corresponding recognition versions of these optimization problems to be \( \mathsf{NP} \)-complete.

The recognition versions are easily derived from the optimization problems by use of an additional parameter \( A \) in replacing a 'minimum' cost by a cost of no more than \( A \) (\( \leq A \)) and a 'maximum' cost by a cost of at least \( A \) (\( \geq A \)). The point to observe about the correspondence is that as long as the optimal cost function is easy to evaluate (here simply the sum of distances), the recognition version is reducible to the corresponding optimization problem and thus no harder. For all we need do is obtain the optimal cost and compare it to the given bound \( A \). Thus if we show that the recognition version of an optimization problem is \( \mathsf{NP} \)-complete, we would know that the optimization version is at least as hard (\( \mathsf{NP} \)-hard).

To show the recognition versions of the above optimization problems to be \( \mathsf{NP} \)-complete we must formulate them in a more suitable manner. We assume that the set of destination points \( T \) are given as a set of integer coordinate pairs. Furthermore we assume that the set \( T = \{p_1, \ldots, p_m\} \) is a multiset with \( \mu_i \) points in \( T \) with exactly the same coordinate \( p_i \), conforming to a destination point \( p_i \) having an integer weight \( \mu_i \). From the optimization problems \( PP_i \), we obtain the corresponding recognition problems \( PP_i \).

Given the multiset \( T \) of destination points as specified before and integers \( k, m \) and \( L \).

\( PP_1 \)

Is there a set \( KS = \{s_1, \ldots, s_k\} \) of \( k \) sources in the plane such that the sum of the distances between the destinations in \( T \) and the sources closest to them is \( \leq L \).

\( PP_2 \)

Is there a subset \( T' \subseteq T \), \( |T'| \geq m \), such that for a set \( KS = \{s_1, \ldots, s_k\} \) of \( k \) sources in the plane, the sum of the distances between the destinations in \( T' \) and the sources closest to them is \( \leq L \).

\( PP_3 \)

Is there a set \( KS \), \( |KS| \leq k \), of sources in the plane, such that the sum of the distances between the destinations in \( T \) and the sources closest to them is \( \leq L \).

One can also formulate the corresponding problems \( QQ's \) and \( RR's \) with the location of sources restricted to finite sets, as specified before.

Lemma 2.1 : \( PP_1 \) reduces to \( PP_2 \). Further, \( PP_1 \) reduces to \( PP_3 \). Similar results hold for the problems \( QQ's \) and problem \( RR's \).

Proof : \( PP_1 \) reduces to \( PP_2 \) for \( T' = T \) and \( m = n \). Further, \( PP_1 \) directly reduces to \( PP_3 \) since if less than \( k \) sources satisfy the limit \( L \), \( k \) sources would definitely do so. \( \Box \)

The discrete problem, \( RR_1 \) was shown to be \( \mathsf{NP} \)-complete for the (integerized) Euclidean, \( L_2 \) distance metric in \([Pa81]\). Since the \( RR \) problems are restricted versions of the corresponding \( QQ \) problems, it follows that the finite solution space problem, \( QQ_1 \) is also \( \mathsf{NP} \)-complete for the \( L_2 \) distance metric. In \([MS84]\), the infinite solution space
Then the shadow of the configuration, corresponding to the elements of $M_i$, consists of $i$ rows $y_1, \ldots, y_i$ of height $1$, arranged parallel to each other, with all there is an extra cover for $\mathcal{C}$ of $y_i$. Each of the points $y_i$, $y_j$ and $y_k$ has a distance with a chain $c$ of the $l^2$-norm. For the reduction, given any instance $(M, \mathcal{C})$ and the family $\mathcal{C}$, we define a distance metric on the elements of $\mathcal{C}$.

**Theorem 2:** Problem $1^\mathcal{C}$ having a direct polynomial reduction, is $\mathcal{NP}$-complete.

We have shown that each of the above optimization problems is in $\mathcal{NP}$. It is left to show that each of these optimization problems is $\mathcal{NP}$-complete. From the earlier discussion we know that $\mathcal{NP}$-completeness of the above problems is not enough for the above problems to be $\mathcal{NP}$-complete, but also $\mathcal{NP}$-hard. Hence, we can define a unique problem $1^\mathcal{C}$ for which the problem $1^\mathcal{C}$ is $\mathcal{NP}$-complete.

Let $(M, \mathcal{C})$ be a given instance of $1^\mathcal{C}$. We define a unique problem $1^\mathcal{C}$ for which the problem $1^\mathcal{C}$ is $\mathcal{NP}$-complete. We define a unique problem $1^\mathcal{C}$ for which the problem $1^\mathcal{C}$ is $\mathcal{NP}$-complete. We define a unique problem $1^\mathcal{C}$ for which the problem $1^\mathcal{C}$ is $\mathcal{NP}$-complete.
Then there exists a subset $KS \subseteq T$ of size $k$ with cost $L$ or less iff $F$ contains an exact cover $C$ of $U$. Assume such a $KS$ exists. Then for the cost to be $L$ or less the optimal allocation is as specified above. Each row is then grouped by solution 1 or 2. Let row $R_j$ grouped as solution 1 mean that $S_j \in C$ where $C$ is the claimed exact cover.

Again, for the cost to be at most $L$, at least $n$ rows must be grouped by solution 1.

For $R_j$ grouped by solution 1 consider the $i^{th}$ group where $u_i \in S_j$. Then both $w$ (above) and $y$ (below) positions are occupied by a point. Since positions $w$, $x$, $y$ or $z$ can charge only 2 to the cost in $L_3$ of $L$, the $w$ and $y$ must connect to their corresponding $q$ sources. This further implies that the $x$ or $y$ point of $q_j$ must be picked up by their lower sources in $R_{j-1}$ and $w$ or $z$ point of $q_j$ by their upper source. The same argument repeats and one notes that this change of upper to lower occurs only once per column, with the $i^{th}$ group of any row $R_k$: $k < j$, must have a lower source while for $k > j$, must have an upper source. Moreover $R_j$ causes this change to all three columns corresponding to the three elements $u_i \in S_j$. Also there are no overlaps in the sets $S_j$ of $C$, since if $u_i \in S_j$ then by the crucial construction of the window, the positions $w$ or $y$ are 2 away just from the solution 2 source (appropriate upper or lower) and if linked up this way, implies $R_j$ is grouped as solution 2 which means that $S_j \notin C$. Hence $C$ contains $n$ sets without overlap and so is an Exact Cover.

Conversely, assume there is an Exact Cover $C$, then there exists a solution $KS$ having $k$ sources of cost at most $L$, by allocating $3n + 2$ sources per row, 1 for each $q'$ pair, with $R_j$ grouped by solution 1 if $S_j \in C$ and solution 2 otherwise. Also for each (unique) $S_j \notin C$ and $u_i \in S_j$, let the $i^{th}$ group of $R_k$ for $k < j$ have a lower source.

The window is shown above in detail. The spots $x, y, w, z$ are not points of $T$ but only possible positions of points. For each window, one of $x, y$ and one of $w, z$ positions is occupied with points of weight $n^2$; $z$ is occupied iff $u_i \in S_{j-1}$, $y$ iff $u_i \in S_j$. Similarly $w$ iff $u_i \in S_{j-1}$ and $z$ iff $u_i \in S_j$. (We use fractional weights with the assumption that all coordinates as well as the limit $L$ will eventually be multiplied by a sufficiently large integer and rounded).

Define $k = t(3n + 2) + 3n(t - 1)$. The first term provides enough sources for all the $t$ rows and the second term, one source for the $q, q'$ pair in each window. $L$ consists of three components $L_1 + L_2 + L_3$. $L_1 = t(2t1.5 + 3n(4.0)) - 2n$ and comes from the $t$ rows. $L_2 = 3n(t - 1)$ and comes from the cost due to the $q$ or $q'$ pair, from the $3n(t - 1)$ windows at a cost of 1 per pair. $L_3 = 12n(t - 1)/n^2$ is the cost of connecting each of the $8n(t - 1)$ points $x$ or $y$ and $w$ or $z$ to the closest $q$ or $q'$ point, always 2 away.
The number of points in region $y \geq 0$ and $y < 0$ is at most twice the number of points in region $y \leq 0$. Hence, if we consider the number of points in region $y \leq 0$, then we can use the fact that the number of points in region $y \geq 0$ and $y \leq 0$ is at most twice the number of points in region $y \leq 0$. Similarly, if we are given the points in a direction parallel to the $y$-axis, then we can use the fact that the number of points in region $y \geq 0$ and $y > 0$ is at most twice the number of points in region $y \leq 0$.

The distance from the set of destination points is $\geq \sqrt{2}$. Let $u$ be any point in the set of destination points. Then the distance from $u$ to any point in the set of destination points is $\geq \sqrt{2}$.

**Lemma 2.2:** In a planar graph, a source point in the plane is at least a certain distance from any point in the set of destination points.

**Proof:**

Let $u$ be any point in the set of destination points. Then the distance from $u$ to any point in the set of destination points is $\geq \sqrt{2}$. Let $u$ be any point in the set of destination points. Then the distance from $u$ to any point in the set of destination points is $\geq \sqrt{2}$.
Theorem 2.4: The problem \( PP_1 \) having the entire Euclidean plane as a solution space, is \( NP \)-complete for the \( l_{\infty} \) distance metric.

Proof: For the completeness, we use the same configuration of destination points \( T \) as in Theorem 2.2. For this \( T \), we claim that there exists a set of \( k \) points, \( KS \), in the plane with cost \( \leq L \) iff there exists a subset of \( k \) points of \( T \) with cost \( \leq L \). The if-part is trivial. For the only-if part, assume there exists a set of \( k \) points \( KS \) in the plane. Note that for a known allocation the geometric location-allocation problem reduces to the location of \( k \) single source problems. The allocation for the above configuration is clearly determined by design; \((3n+2)\) sources per row and 1 per window.

For each of the rows and windows, an application of Lemma 2.3 shows that there exists a corresponding subset of \( k \) points of cost \( \leq L \). □

We now prove a result similar to Lemma 2.3, for locating sources under the \( l_1 \) distance metric. This result previously appears in [IPT80]. However, we provide a much simpler proof of this, in a way which suffices for our goal of proving \( QQ_1 \), for the \( l_1 \) distance metric, to be \( NP \)-complete. Define the \( l_1 \) grid points of the set \( T \) of destination points to be the points which have an \( x \)-coordinate equal to the \( x \)-coordinate of any point \( p \in T \) and a \( y \)-coordinate equal to the \( y \)-coordinate of any point \( q \in T \), as in figure 2.5.

Lemma 2.5: In locating a source point in the plane so that the sum of the \( l_1 \) distances from the destinations \( T \) is \( \leq L \), one could as well let the source point be an element of the set \( S \), one of the \( l_1 \) grid points of the set \( T \).

Proof: Let us assume that for the source point \( p \) as shown in figure 2.6, the sum of the \( l_1 \) distances from the \( T \) destination points is \( \leq L \).

Then there must exist \( l_1 \) grid lines \( a \) and \( b \) such that point \( p \) lies between them, but
From \( x \neq x' \), then \( x, x' \notin \mathbb{N} \), \( x, x' \notin \mathbb{N} \), where \( x, x' \notin \mathbb{N} \), and \( x, x' \notin \mathbb{N} \), and \( x, x' \notin \mathbb{N} \). Thus, \( (x, x') \in \mathbb{N} \), when \( (x, x') \notin \mathbb{N} \), and \( (x, x') \notin \mathbb{N} \). The same holds for \( (x, x') \in \mathbb{N} \).

Consider the two functions \( f(x) \) and \( g(x) \), where \( f(x) \) is an encoded function of one

\[ f(x) = \begin{cases} 1 & \text{if } x = x' \\ 0 & \text{otherwise} \end{cases} \]

Then \( f(x) \) is in \( \mathbb{N} \), and \( g(x) \) is a function \( : \mathbb{N} \to \mathbb{N} \).
Theorem 2.7: All the problems PP'₁, QQ'₁ and RR'₁, for each of the three distance metrics \( l_1, l_2 \) and \( l_\infty \), are strongly NP-complete.

Proof: The magnitude of the largest number occurring in any instance \( I \) of the problems PP'₁'s (similar for QQ'₁'s and RR'₁'s) for the \( l_\infty \) distance metric, is determined by either the coordinate pairs of the destination set \( T \) or the parameters \( k, m \) or \( L \). The parameters \( k \) and \( m \) are both integer values less than \( n \). The constructions in Theorem 2.2 and Lemma 2.3, show the strong NP-completeness of PP'₁, QQ'₁ and RR'₁, since what is exhibited is a bounded polynomial transformation from Exact Cover (a strong NP-complete problem [GJ79]). The lengths of the two problems are polynomially related and a polynomial of \( n \). Further the function (the iff reduction mapping) is polynomial time computable. The integer coordinate pairs of the points of \( T \) are bounded by \( O(n) \) and the value of \( L \) can be seen to be bounded by \( O(n^3) \). Hence the maximum value occurring in the construction of the geometric location-allocations problems, is bounded by a polynomial in \( n \) and hence bounded by a polynomial of the maximum value and length of an instance of the Exact Cover problem (satisfying the conditions of the pseudo-polynomial transformation).

The reduction of [MS81] for the NP-completeness of problem PP'₁ for the \( l_1 \) and \( l_2 \) distances, is a bounded transformation from 3-Sat (strongly NP-complete [GJ79]) and hence the strong NP-completeness for these problems follows as above. The strong completeness of the problems QQ'₁ and RR'₁ for the \( l_1 \) metric (Theorem 2.6) and for the \( l_2 \) metric [Bar81] follows again from their transformation from the Exact Cover problem.

The problems PP'₂ and PP'₃ (similarly for the corresponding QQ and RR problems), have \( m \) and \( k \), respectively, as their decision parameters and are actually number problems, consequently strongly NP-complete. The strong completeness of these problems also follow from the direct transformation from the above PP'₁ problems. (Lemma 2.1) ∎

We can now finally show that our (decision) optimization problems \( O_i \)'s are \( \Delta_2^P \)-proper for infinite, finite and discrete solution sets. [LM81] first showed how pure combinatorial optimization problems can be proven to be \( \Delta_2^P \)-proper. We show how it applies to our case of geometric optimization problems.

Lemma 2.8: The optimization problems \( O_i \)'s are in \( \Delta_2^P \).

Proof: A problem \( P \) is in \( \Delta_2^P \) if \( P \leq_T^P \) (Turing reduces) to \( Q \), where \( Q \) is in \( \Delta_2^P \).

For each of the \( O_i \)'s, where the sources are located in the entire plane, we show that a Turing reduction exists to the corresponding recognition NP-complete versions PP'₁'s and [MU74] (For those \( O_i \)'s where the sources are located on a discrete set \( S \) or on the set of destination points \( T \), the corresponding NP-complete versions QQ'₁'s and RR'₁'s are used respectively). For \( O_i \), the value of parameter \( L \) in any instance \( I \) of the problem, lies in the range \( 0 \) to \( \epsilon \cdot \text{Length}[I] \), for some constant \( \epsilon \). By a simple binary search one can find the optimum value of \( L \) in at most \( \log \epsilon \cdot \text{Length}[I] = O(\text{Length}[I]) \) calls of the PP'₁ oracle. Since \( \text{Length}[I] \) is some polynomial in \( n \), the reduction is polynomially time bound.

For problems \( O_2 \) and \( O_3 \) with their respective parameters \( m \) and \( k \), the parameter values range from \( 1 \) to \( n \) and so a sequential search suffices. ∎
of the worst situation. A common real world situation that might be formulated with a minimax objective is one of locating health clinics so that the maximum distance a patient must travel to a clinic is minimized. Another example concerns the placement of fire stations in a large metropolitan area such that the maximum distance between any location within the city and the nearest fire station is minimized.

Under this minimax location objective it is possible to distinguish two basic approaches that have been taken in the literature on facility location. The first suggests that a location site may be selected anywhere in the area of interest on the plane, giving an infinite number of possible location sites. The second approach considers only a finite number of known sites as feasible and models the constraints imposed on the possible location of sources, ensuring that undesirable and impractical locations need not be considered.

In this paper we analyze the complexity of certain geometric optimization problems which arise frequently in the above application areas. We are given a set $T = \{(x_i, y_i), i = 1, n\}$, the location of $n$ fixed destination points (destinations) in the plane and need to locate a number of points (sources) to service the destinations in a way which behooves the application on hand. Three different optimization questions are considered as $P_1$, $P_2$ and $P_3$ below, each under a minimax optimizing objective. Note that in the case of locating multiple sources, the optimal allocation of the destinations to the sources must also be determined - hence the name location-allocation problems [Cot65]. On the assumption that each destination can be completely serviced by a single source, clearly in the optimal solution each destination is allocated to its closest located source.

Given the set $T$, as specified before, of $n$ destinations in the plane

$\{P_1\}$ Locate $k$ points (sources) so as to minimize the maximum of the weighted distances between the destinations and the sources closest to them.

$\{P_2\}$ Locate $k$ points (sources) so that for a maximum number of destinations the weighted distances of these destinations from their closest sources does not exceed a prescribed limit $R$.

$\{P_3\}$ Locate a minimum number of points (sources) so that the maximum of the weighted distances of the destinations and their closest sources does not exceed a prescribed limit $R$.

The weighted distances mentioned above come from a weight $w_j$ assigned to each destination $j$ and is some measure of the special cost of serving destination $j$ in travelling from its closest source. It may be interpreted as the importance or probability of an emergency occurring and therefore will often be a function of the size of the destination. However in the following problems we assume that all weights are equal (similar to assuming that $w_j = 1$, for $j = 1, n$) and show that even for this restricted case these above problems are quite difficult.

Problems $P_1$, $P_2$ and $P_3$ allow location of the sources to be anywhere in the plane. Let problems $Q_1$, $Q_2$ and $Q_3$ correspond to problems $P_1$, $P_2$ and $P_3$ respectively, with the location of the sources being restricted to a finite discrete set $S$ of possible locations in the plane and of size polynomial in $n$. Note that all the above questions are of the form 'Locate the optimal ...'. However, they may also be stated in the traditional optimization version of 'Find the optimal ...', wherein we find the optimal number / size of the
2.3.1. The Boolean difference metric

Let $A$ and $B$ be sets in $\mathbb{R}^n$. The Boolean difference $A - B$ is defined as the set $A \setminus B$, which contains all elements of $A$ that are not in $B$.

2.3.2. The Boolean symmetric difference metric

Let $A$ and $B$ be sets in $\mathbb{R}^n$. The Boolean symmetric difference $A \Delta B$ is defined as the set $(A \cup B) \setminus (A \cap B)$, which contains all elements that are in either $A$ or $B$, but not in both.

2.3.3. The Boolean distance metric

Let $A$ and $B$ be sets in $\mathbb{R}^n$. The Boolean distance $d(A, B)$ is defined as the minimum number of elements that need to be added to or removed from $A$ to obtain $B$.
anywhere in $S$ can cover? 

\{ QC_3 \} 

Is $k$ the minimum number of $R$-circles locatable anywhere in $S$ to cover the $n$ points of $T$? 

For circles (circular discs) an alternate though similar set of optimization questions as above, may be asked. The piercing number for a set of circles is the number of 'needles' required to pierce all the circles of the system [HID64].

Given a set $C$ of $n$ fixed $R$-circles in the plane,

\{ DC_2 \} 

Is $m$ the maximum number of $R$-circles of $C$ that $k$ needles can pierce? 

\{ DC_3 \} 

Is $k$ the minimum number of needles required to pierce all $n$ $R$-circles of $C$? 

Note that in these problems we optimally locate 'needles' (points), in order to tag each of the given circles.

**Theorem 2.12**: The problems $DC_2$ and $DC_3$ are dual problems of $PC_2$ and $PC_3$ (respectively) and are polynomial-time transformable to each other.

**Proof**: The duality arises from the fact that if the $k$ $R$-circles with centers $\{ r_i \}$, $i = 1, k$, have a common intersection point then they can all be pierced by a single needle. Furthermore, a single $R$-circle centered on any point of the common intersection can cover the $k$ centers, $\{ r_i \}$, $i = 1, k$.

Given a set $T = \{ n$ points in the plane$, \}$, obtain the set $C$ of $n$ circles of radius $R$ with centers being the $n$ points of $T$. Then $m$ is the maximum number of $R$-circles of $C$ that $k$ needles can pierce iff $m$ is the maximum number of points of $T$ that $k$ $R$-circles can cover. The proof follows, since for each subset $S$ of $C$ of $R$ circles that a needle pierces, an $R$-circle centered on the piercing point of the needle can cover the centers (members of set $T$), of the $R$-circles of set $S$. Conversely, for each subset $S$ of points, $S \subseteq T$, that are coverable by an $R$-circle, a needle pierces the subset of $C$ of $R$-circles having $S$ as their centers.

Similarly $k$ is the minimum number of needles required to pierce all $n$ $R$-circles of $C$ iff $k$ is the minimum number of $R$-circles required to cover the $n$ points of $T$. \ --- \}

We now show that the problem $PC_3$ of locating the minimum number of $R$-circles in $\mathbb{R}^2$ to cover all the $n$ demand points is $\mathsf{DP}$-complete by reducing $\langle \text{Sat}, \text{UnSat} \rangle$, a known $\mathsf{DP}$-complete problem [PY82], to it. We adapt certain constructions previously specified in [FPT81]. Next we show that all the above remaining problems are $\mathsf{DP}$-complete by a series of polynomial time reductions. To show a problem to be complete for this class $\mathsf{DP}$ we differ from [PY82] in that we use polynomial-time positive (disjunctive) reductions [LLS75] as opposed to polynomial-time many one reductions.

In a simplistic fashion any form of polynomial-time oracle-reducibility (which includes both positive and Turing reductions), is a Boolean formula of a polynomial number of queries to the oracle. The essential restriction for positive reductions is that the Boolean formula is a positive formula. Sufficient to our purpose here, the Boolean formula is positive if it only contains disjunctive (V) and conjunctive (A) Boolean connectives. Furthermore, in positive reductions, like in other truth table reductions, one is restricted to a prespecified list $f(z)$ of queries, based on input $z$, from which alone one
under a polynomial-time positive (disjunctive) reduction from (Sat, Unsat). \(\square\)

**Theorem 2.14**: The problem \(PC_2\) of locating \(k\) circles in \(R^2\) to cover a maximum number of the given points is \(\text{DP complete}\).

**Proof**: The problem is in \(\text{DP}\) since it can be rephrased as before, as the conjunction of a predicate in \(\text{NP}\) and a predicate in \(\text{Co NP}\) \(\exists\ p_1...q_k \text{ in } E^2\) \(\|R\) circles with centers at \(p_1...q_k\) cover \(m\) points\(\) \(\land\ \forall q_1...q_k \text{ in } E^2\) \(\|R\) circles with centers at \(q_1...q_k\) cover \(\leq m\) points.\(\)

To prove the completeness we again reduce (Sat, Unsat) to \(PC_3\), using polynomial-time positive (disjunctive) reductions in a way very similar to above.

Starting from \((F_1, F_2)\) we construct two separate sets of points \(S_1\) and \(S_2\) in the plane such that for \(i=1,2\), \(R\) circles are required to cover all the \(n_i\) points in \(S_i\) if \(F_i\) is satisfiable. Further, if \(F_i\) is not satisfiable, \(k_i\) \(R\) circles can cover at least \(n_i - c_i\) points and at most \(n_i + c_i\) points of \(S_i\), where \(c_i\) is the number of clauses in the \(\text{CNF}\) form of \(F_i\).

Now construct \(c_2+1\) additional copies of the set of points \(S_1\). We now have \((c_2+1)\) copies of sets of points \(S_1\) and a single set of points \(S_2\). It is important to note why \((c_2+1)\) copies of \(S_1\) are required. Let \(k=(c_2+1)k_1+k_2\). It is not hard to see that \(m\), the maximum number of points that can be covered by \(k\) circles of radius \(R\), satisfies \((c_2+1)n_1 + n_2 \leq m \leq (c_2+1)n_1 + n_2 + 1\) if \(F_1\) is satisfiable and \(F_2\) is not satisfiable.

Since this is a disjunction of at most \(c_2\) calls of \(PC_2\) problem \(PC_2\) is \(\text{DP complete}\).

\(\square\)

**Corollary 2.14.1**: The dual problems \(DC_2\) and \(DC_3\) are also \(\text{DP complete}\).

**Theorem 2.15**: The problem \(PC_4\) of locating \(k\) equal sized circles of minimum radius to cover all the given \(n\) points is \(\text{DP complete}\).

**Proof**: The problem is in \(\text{DP}\), when \(R\) is restricted to integers\(^4\), since it can be rephrased as before, as the conjunction of a predicate in \(\text{NP}\) and a predicate in \(\text{Co NP}\) \(\exists[p_1...q_k] \text{ in } E^2\) \(\|R\) circles with centers at \(p_1...q_k\) cover \(n\) points\(\) \(\land\ \forall q_1...q_k \text{ in } E^2\) \(\|R\) circles with centers at \(q_1...q_k\) cover \(< n\) points.\(\)

To prove it complete we show that \(PC_3\) polynomial-time positive reduces to \(PC_4\).

We construct a set \(S\) of the radii of all possible circles which minimally cover \(n\) points in the plane. Since the minimum enclosing circle for a set of points is defined by exactly two or three of the points, the total size of \(S\) is at most \(\binom{n}{2} + \binom{n}{3}\) which is \(O(n^3)\). We claim that \(k\) is the minimum number of \(R\) circles that cover all \(n\) points off for some \(S, \leq R, s\) is the minimum radius of \(k\) circles to cover all \(n\) points and for some \(S, s > R, s\) is the minimum radius of \(k\) circles to cover all \(n\) points. The proof is straightforward and follows from the definitions of the two problems \(PC_3\) and \(PC_4\).

Again since we have a conjunction of two sets of disjunctive calls of \(PC_4\) [at most \(O(n^3)\) calls], we have a polynomial-time positive reduction from \(PC_3\) to \(PC_4\). \(\square\)

\(^4\) Otherwise the problem appears to be \(\text{DP hard}\) when \(R\) is in general, a real number.
...
2.3.2. The \( (l_p, 1 \leq p \leq \infty) \) distance metrics

With the general \( l_p \) distance metrics each of the above location-allocation optimization problems reduces to the placement of equal sized\(^5\) geometrical figures.

\[ \text{Figure 2.7.} \]

\(^5\) Again assuming equal weights for the destinations results in our considering each of the previous problems \( P \)'s and \( Q \)'s for equal sized geometrical figures. The unit areas are given by \( |x|_p + |y|_p = 1 \)

all oriented the same way, and with the intersection points of their \( xx' \) and \( yy' \) axes corresponding to the location of the sources.

For the rectilinear \( l_1 \) distance metric the geometric figures involved in our above optimization problems \( P \)'s and \( Q \)'s are equal sized diamonds (squares rotated by 45°) of half-diagonal length \( R \) instead of \( R \)-circles. For the infinity \( l_\infty \) distance metric the optimization problems reduce to placement optimization problems of equal sized squares of half-edge length \( R \), having sides parallel to the respective coordinate axes. Also a diamond (square) locatable anywhere in \( S \) or \( E^2 \) means the intersection point of the diagonal of the diamond (square) can be any point in the finite discrete set \( S \) or the Euclidean plane respectively.

For all these problems both membership and completeness for the class of \( D^P \) carry over in a fashion quite similar to the proofs of Theorems 2.13 to 2.18. Each of the constructions used here as well as the adapted constructions of [F15,T51], can be modified in a direct fashion for these geometric figures with fixed orientations. Thus we have the following result.

**Theorem 2.19**: The above \( P_1, P_2, P_3 \) and \( Q_1, Q_2, Q_3 \) optimization problems, under the general \( l_p \) distance metric, are solvable as placement optimization problems, (decision versions), for the corresponding geometric figures are all \( D^P \) complete.

2.3.3. Arbitrary distance metrics

Between two points \( p \) and \( q \) in the plane, the various properties of a distance metric \( d(p,q) \) are:

1. **translation invariance**: \( d(p,q) \) is a function of the relative positions of \( p \) and \( q \).
2. **symmetry**: \( d(p,q) = d(q,p) \).
3. **triangle inequality**: \( d(p,r) \leq d(p,q) + d(q,r) \).
4. **positivity**: \( d(p,q) \geq 0 \).
5. **homogeneity**: For \( a = (0,0) \), \( p = (x,y) \) and \( q = (ax, ay) \), \( d(a,q) = ad(a,p) \).

Each of the \( l_p, 1 \leq p \leq \infty \), distance metrics satisfy the above properties. The Euclidean \( l_2 \) distance metric is the only distance which is also invariant under rotation. That is the reason why in Section 2.3.2, above, a specific fixed orientation was
(2) Change properties of the class A with many one

reduction

(1) Are the above geometric optimization problems A-complete under many one

reduction?

[Diagram: A geometric figure with labels and points marked]
2.3.4. Appendix A

The figure provides an overall view of the construction. Each of the lines is a wire, a set of points arranged so that any pair of successive points along a wire can be covered by one circle and any given circle can cover at most two points along a wire. Corresponding to each variable in the formula is a closed loop of wire which contains an even number of points. There are two possible coverings, one of these corresponds to an assignment of true and the other designates false.

On embedding the construction in the plane, a suitable crossover is designed for the loops that cross on another. At the crossover the two loops share a circle such that the value assignments of both loops are propagated independently through the crossover.

For each of the clauses there is a point located in a region in which the three variable loops corresponding to the literals of the clause are brought into close proximity. The loops are arranged so that the clause point can be connected 'for free', if at least one of the three wires is in the state corresponding to the truth of the literal in the clause. (If the literal is a negated variable, the variable loop is adjusted so that the clause point can be covered for free by the covering corresponding to the assignment of false to the variable). If none of the literals is true then an extra circle is needed to cover the clause point. If all the clauses are simultaneously satisfied then no additional circles are necessary to cover the clause points. The construction is therefore coverable by \( k \sigma \) circles if the formula is satisfiable.
CHAPTER 3

3.1 Extended

EFFICIENT ALGORITHMS AND LOWER BOUNDS
3.2. Geometric Location

In describing the wide variety of queuing problems, Wagner [see FW74] states "What is said about birds is also true of waiting line models: their variety and number seem infinite." The same is becoming true for geometric optimization problems. Such problems frequently arise in facility location, clustering analysis, VLSI layout design, and other Cad/Cam, robotics applications.

In this paper we consider the problem of determining the location of a single new facility (source) with respect to a number of existing users (destinations). The location we seek is that which minimizes an appropriately defined total cost function. We consider the cost to be proportional to the distance. A number of single facility location problems exist and are amenable to the analysis presented here. Some of the more traditional examples are the location of a

1. A new lathe in a machine tool workshop.  \textit{(minsum)}
2. A welding robot in an automobile manufacturing plant. \textit{(minsum)}
3. A new warehouse relative to production centers and customers. \textit{(minsum)}
4. A hospital or a fire station in a metropolitan area. \textit{(minmax)}
5. A cleansing pump in a chemical processing setup. \textit{(minmax)}
6. A component in a VLSI design chip. \textit{(minsum)}
7. A water fountain in an office building. \textit{(minmax)}

These examples suggest a reason for the interdisciplinary interest in facility location.

We also consider some related geometric optimization problems that stem from the above application areas and devise both efficient algorithms and non-linear lower bounds for them. Since nearly all of these problems are recast in a mathematical framework bordering on Euclidean geometry these optimization problems are mathematically intriguing in their own right.

The discrete optimization criteria \textit{minmax} (minimizing the maximum cost) and \textit{minsum} (minimizing the sum of costs), are two of the more important location objectives that arise in numerous applications as above. The distance metrics used \textit{Rectilinear} (\textit{L}_1) and \textit{Euclidean} (\textit{L}_2) reflect the appropriate restrictions on 'travel' between the sources and destinations, for the application under consideration. It is also possible to distinguish two basic approaches that have been taken in the literature on facility location. The first suggests that a location site may be selected anywhere in the area of interest on the plane, giving an \textit{infinite} number of possible location sites. The second approach considers only a \textit{finite} number of known sites as feasible and models the constraints imposed on the possible location of sources, ensuring that undesirable and impractical locations need not be considered.

Here we consider two different point geometric location problems in the plane.

Given a set \(T = \{(x_i, y_i), i = 1, n\}\), the location of \(n\) fixed destination points \(\text{destinations}\) in the plane, under \textit{minmax} optimization, we have the problems.

\(P_1\): Locate a point (source) so as to minimize the maximum of the distances between the source and the destinations.

\(P_2\): Locate a point (source) so that for a maximum number of \textit{destinations} the distances of these destinations from the source does not exceed a prescribed limit \(R\).
The theorem point function approach for the first metric can be summarized in a simple manner. Given the theorem point function approach for metric functions,

$\text{Theorem point function approach formula}$

where $F(x,y)$ is the theorem point function approach for metric $x$, $y$ is the metric approach, and $c$ is a constant.

The theorem point function approach for metric $x$ is defined as

$\text{Theorem point function approach for metric } x = f(x) + g(y)$

where $f(x)$ is the function associated with metric $x$ and $g(y)$ is the function associated with metric $y$.

To determine the theorem point function approach for a given metric $x$, we first compute the function $f(x)$ and then apply the function $g(y)$ to compute the theorem point function approach for metric $x$.

In practice, the theorem point function approach for metric $x$ can be applied to determine the theorem point function approach for a different metric $y$ as

$\text{Theorem point function approach for metric } y = h(y)$

where $h(y)$ is a function associated with metric $y$.

The theorem point function approach formula can be used to compute the theorem point function approach for a given metric $x$. Thus, for the theorem point function approach formula

$\text{Theorem point function approach formula}$

the theorem point function approach for metric $x$ can be computed as

$\text{Theorem point function approach for metric } x = f(x) + g(y)$

where $f(x)$ is the function associated with metric $x$ and $g(y)$ is the function associated with metric $y$.
$O(n \log n)$ time [LW80]. The farthest point diagram can be constructed in an analogous procedure.

(P2) Locate a source anywhere in the plane, so that a maximum number of the $n$ destinations have their distances from the source $\leq R$.

Euclidean distances. $O(n^2 \log n)$ time. $\Omega(n \log n)$

(Actually $O(m \log n)$, where $m = \text{maximum number of points covered by a circle of radius } 2R$).

The problem reduces to finding the maximum number of points that can be covered by a circle of radius $R$ ($R$ circle). A dual problem can also be stated as follows, given $n$ $R$ circles in the plane, 'Find the maximum value of $k$ for which there exists a $k$-intersection, the common intersection of $k$ circles'. The problems are dual since a solution for one is a solution of the other and vice versa.

A $\Omega(n \log n)$ lower bound under the linear decision tree model is shown here by a reduction from the set element uniqueness problem which has been shown to have an $\Omega(n \log n)$ lower bound under the same model, in [DL78], and also under the algebraic computation tree and bounded degree algebraic decision tree models, in [Be83]. The reduction follows directly, wherein the elements of the set are mapped to points in the plane which for a limit $R = 0$ returns a maximum of $1$ if the elements of the set are distinct.

In 1-dimension where circles of equal radius $R$ degenerate to equal line segments (or intervals) of length $R$ on a line the problem can each be solved in $O(\log n)$ time by sorting the left-endpoints of the $n$ intervals and scanning the points on the line in linear time. The lower bound of $\Omega(n \log n)$ also applies to the 1-D problem in an identical fashion to above, by mapping the elements of the set to points on a line and asking the question for a limit $R = 0$. Again the answer is a maximum of 1 iff the elements of the set are distinct. Hence for 1-D the problem $P_2$ is $O(n \log n)$.

Lemma 3.1: There are at most 2 circles of a given radius $R$ ($R$ circle) which pass through 2 given points.

Proof: If we travel clockwise along the circumference of the circle passing through the two points, then either a minor arc connects the two points or a major arc: giving rise to at most two positions of the $R$ circle. The case of a unique $R$ circle is when the two points are exactly $2R$ apart and form the diameter of the $R$ circle.

Theorem 3.2: In 2-D the problem $P_2$ can be solved in $O(n^2 \log n)$ time.

Proof: For each of the centers $p_i$, $i = 1...n$ we do the following

(1) Let $M_i$ be the set of points within the euclidean distance $\leq 2R$ of $p_i$. [The size of $M_i$ is at worst $n$]. Define the "\( \approx \)" order between two points $p_j$ and $p_k$ to be, $p_j \approx p_k$ if there is an $R$ circle (circle of radius $R$), passing through points $p_j$ and $p_k$ such that the point $p_j$ is 'below' the arc $p_jp_k$. Sort the set of points $M_i$ according to this ordering relation. [This is similar to sorting by polar angle with $p_i$ as origin, wherein the points $p_j$ and $p_k$ are compared with respect to the line $p_jp_k$ as opposed to the arc of a circle of radius $R$ as is done above. Thus the definition of $p_j$ 'below' are $p_i \approx p_k$ is: On rotating the arc of a circle of radius $R$ in an anti-clockwise direction about a point $p_i$, the point $p_j$ intersects the arc 'before' the point $p_k$.] On the basis of Lemma 3.1 we can test the "\( \approx \)" relation in $O(1)$ time giving a worst case $O(n \log n)$ sort-
The lower bound on the reduction from the set cover problem is at least $(\log n + \log \log n)$ times larger than the lower bound on the independent set problem.

**Lemma 1:** For points in $\mathbb{R}^n$, the problem of finding the minimum number of points that can be covered by a hyperplane is computationally hard.

**Theorem 2:** For points in $\mathbb{R}^n$, the problem of finding the minimum number of points that can be covered by a hyperplane is computationally hard.

Thus, the problem of finding the minimum number of points that can be covered by a hyperplane is computationally hard.

The time taken to find the minimum number of points that can be covered by a hyperplane is computationally hard. The time taken to find the minimum number of points that can be covered by a hyperplane is computationally hard.
Rectilinear distances, $O(n \log n)$ time.

Actually $O(n \log n + k)$, where $k$ = maximum number of points reported to lie within a diamond of half-diagonal length $R$. The problem is solved by a 'batched range search' technique of [BW80]. For each of the $n$ destination points we have a square with edge length $\sqrt{2}R$, whose 'center' coincides with the destination point. Then using the range search technique, the set of destination points are organised into a data structure and for each square, queries are made for the list of all destination points that are contained in it. The total time taken is given by the above time bound. The fact that our original problem for the rectilinear distance metric reduces to diamonds of half-diagonal length $R$ is easily settled by rotating the entire coordinate plane containing the destination points by an angle of 45° transforming it to squares of edge length $\sqrt{2}R$.

The lower bound $\Omega(n \log n)$ is by reduction from the set element uniqueness problem as before.

3.2.2. Locating under MINSUM

Such a minimum criterion can be considered as a prototype for many location formulations where some measure of the total or average cost of serving the fixed destinations is to be minimized.

{P*} Locate a source anywhere in the plane, so as to minimize the sum of the distances from the n destinations.

Euclidean distances.

The objective function is strictly convex when the destinations are not collinear and convex in the case of collinearity. In fact the problem in the plane is convex for all $L_p$ distance metrics. Thus a local minimum is a global minimum. However, this somewhat celebrated problem is very difficult and the only known solutions are iterative procedures and heuristic algorithms [FW74],[KP79]. A dynamic programming solution is also given in [Ik65]. In [Chapter 4] we show that any exact method reduces to a minimal polynomial whose roots are not expressible in terms of radicals (and hence the solution would have to be approximated).

The solution to the problem is however simple for some special cases [Ei62]. For example, the problem reduces to the rectilinear case below, when the $n$ destination points are collinear and can be solved in $O(n)$ time. This is also the problem in 1-D.

Rectilinear distances, $O(n)$ time.

In many problem formulations the $L_1$ distance measure is preferred, as perhaps in locating a new warehouse in Manhattan, or a component on a VLSI chip. Also this is the metric of use in various 3-D problems dealing with the installation of pipes, electrical cables in a building. Also more importantly, this measure is much easier than the $L_1$ metric because of its separability.

From the problem formulation, given $n$ destinations $(h_i^x, h_i^y)$, $i = 1, n$, we need to minimize

$$\sum_{(x,y) \in P} \sum_{i=1}^{n} \sqrt{|h_i^x - x| + |h_i^y - y|}$$

It is evident that the objective function is separable so that one can solve the two parts independently. Namely,

$$\min_{x} \sum_{i=1}^{n} |h_i^x - x| \quad \text{and} \quad \min_{y} \sum_{i=1}^{n} |h_i^y - y|.$$ 

This is, to find separate medians for the $x$ and $y$ coordinates of the $\{h_i\}$ points in the plane. This follows from the fact...
The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.

The lower bound is obtained from the following argument. Let
\[ \min_{d \in D} \{ \sum_{i=1}^{n} d(x_i, y_i) \} \]
be the minimum of the distance function over all \( d \in D \). The distance function \( d \) is the sum of the distances from the source to each point in the plane, as defined in Section 2.2.
3.3. Related Geometric Optimization Problems

The problem of intersections of geometric objects has been seen to be related to that of placement of geometric objects with respect to certain fixed points and hence to the problems of geometric location. When the objects become unequal sized, then containment of a smaller geometric object (say a circle) within a larger circle is also considered an intersection. A geometric optimization question that immediately arises, concerning these intersections, is the question of the maximum containment of one object within another.

[Maximum Containment] Given \( n \) circles of unequal radius \( r_i \) in the plane, what is the maximum level \( k \) of containment (maximum number of circles completely within one another)? List the \( k \) circles in the order in which they are contained within each other.

**Theorem 3.4:** The circle containment problem takes \( \Omega(n \log n) \) time.

**Proof:** Given a sequence of elements \( \{x_i\}, i=1..n \), construct \( n \) circles with centers at the origin (say) and having radii \( \{r_i\} \). The maximum level of containment is \( n \) and a listing of the \( n \) circles in the order in which they are contained is a sort of the sequence \( \{r_i\} \).

The problem is \( \Omega(n \log n) \) even in 1-dimension, where given \( n \) intervals on a line of unequal length \( I_i \) we wish to list in order those intervals which enclose each other, conforming to the maximum level of containment. A similar reduction to the one above shows this to be true (construct intervals with their left endpoints all starting at the origin). The containment problem can also be reduced to a directed acyclic graph with vertices \( \{v_i\} \) corresponding to circles \( \{p_i\} \) and directed edges \( <v_i, v_j> \) iff circle \( p_i \) is contained in circle \( p_j \). The circle containment problem then reduces to finding the longest directed path in the acyclic graph.

**Theorem 3.5:** In 1-dimension the containment problem can be solved in \( O(n \log n) \) time and this is optimal.

**Proof:** Again only a smaller sized interval can be contained in a larger interval. For intervals \( I_i, I_j \) with left and right endpoint \( (l_i, r_i) \) and \( (l_j, r_j) \), respectively, \( I_i \) is contained in \( I_j \) if \( l_i \leq l_j \) and \( r_i \leq r_j \). Sort the 2n endpoints of the n intervals. Scan the intervals in a left to right order. On encountering a left point \( l_i \) of an interval \( I_i \) we add this interval name to a separate list \( L_i \) by order of its left endpoint \( l_i \). On encountering a right endpoint \( r_j \) corresponding to a left endpoint \( I_j \) of the interval we determine how many intervals in list \( L_i \) have their left endpoints before \( I_j \). This can be done in \( O(n \log n) \) time via binary search on list \( L_i \). The total time taken is bounded by \( O(n \log n) \).

**Theorem 3.6:** The circle containment problem can be solved in \( O(n^2) \).

**Proof:** Given \( n \) circles with centres \( p_i \) of radius \( r_i \), \( i=1..n \). Sort the radius \( \{r_i\} \) into non-decreasing order. A circle with centre \( p_i \) with smaller radius \( r_i \) is contained in a circle with centre \( p_j \) with larger radius \( r_j \) if \( d(p_i, p_j) \leq r_j - r_i \). Now consider the circles in increasing order of radii, and for each circle check to see if any of the circles before it (with smaller radius) are contained within it. For each circle one
Introduction

ALGEBRAIC COMPLEXITY OF GEOMETRIC OPTIMIZATION
4.2. The Fermat-Weber problem

The Problem

Given \( n \) fixed destination points in the plane with coordinates \( (x_i, y_i) \), determine the location \( (x, y) \) of a single source point such that the sum of the Euclidean distances from this source to each of the destinations is minimized.

\[
\text{minimize}_{x, y} \, f(x, y) = \sum_{i=1}^{n} \sqrt{(x - x_i)^2 + (y - y_i)^2}
\]

The above problem has been considered for quite a long time and has an equally long and interesting history. The problem for the case of \( n = 3 \) was first formulated and thrown out as a challenge by Fermat as early as in the 1600s [Kn07]. Cavalieri in 1647 considered the problem for this case, in particular, when the three points form the vertices of a triangle and showed that each side of the triangle must make an angle of 120° with the given minimum point. Heinen in 1834 noted that in a triangle which has an angle of \( \geq 120° \), the vertex of this angle itself is the minimum point.

Fagnano in 1775 showed that for the case \( n = 4 \) when the four client points form a convex quadrilateral the minimum solution point is the intersection of the diagonals of the quadrilateral. For a non-convex quadrilateral the fourth point which is inside the triangle formed by the three other points, is itself the minimum point.

\[
\begin{align*}
\text{(a) Triangle with angles < 120°} & \quad \text{(b) Triangle with an angle } \geq 120° \\
\text{Figure 4.1.} & \\
\text{(c) Convex quadrilateral} & \quad \text{(d) Non-convex quadrilateral} \\
\text{Figure 4.2.} &
\end{align*}
\]
Further we can show that not only is the solution not constructible by ruler and compass, but that this conclusion is in fact independent of the set of axioms of Euclidean geometry. The construction for the solution points in the case of 3 points is similar to the problem of the three centers of gravity of a triangle. We show below that the non-constructibility of the solution by ruler and compass is a consequence of the independence of the problem from the axioms of Euclidean geometry.
\[
df/\ dy = \sum_{i=1}^{n} (y - b_i) / \sqrt{(x - \alpha_i)^2 + (y - b_i)^2} = 0 \quad (3)
\]

By a process of repeated squaring one can eliminate all the square-roots from the expressions (2) and (3) above. Starting with a sum of \( n \) different square-roots, \( \text{sqrt}(i), i = 1, \ldots, n \), equated to a constant, the technique is to take all terms of \( \text{sqrt}(i) \), for a certain \( i \), to one side of the equation and the remaining terms on the other side, squaring both sides and thereby eliminating \( \text{sqrt}(i) \). Repeating this process by again isolating one of the remaining square-roots and squaring, one is able to eliminate all square-roots from the original equation in a maximum of \( n \) steps.

For the case of 5 points, let \((a_i, b_i), i = 1, 5\) be the given points with integer coordinates. In the following we work over \( \mathbb{Q} \), the field of rational numbers. We make the tacit assumption that the solution does not coincide with any of the destination points and obtain the corresponding polynomials \( f_1(x, y) = 0 \) and \( f_2(x, y) = 0 \) from (2) and (3) respectively, after rationalizing and elimination of the square-roots. The resulting minimal polynomial for the minimization problem is

\[
p(x, y) = f_1(x, y)^2 + f_2(x, y)^2 = 0 \quad (4)
\]

Now that we have the above polynomial \( p(x, y) \) for the problem our first step is to prove it irreducible, over \( \mathbb{Q} \). We show this by substituting constants for \( x, y = c \) and showing that \( p(c, y) \) is irreducible\(^4\). However the fact that the minimal polynomial \( p(x, y) \) is irreducible is important to us only if the line determined by \( x = c \) passes through the solution point of our optimization problem. This is important and the reader should understand the reason. Consequently, we choose symmetric configurations

---

\(^4\) If \( p(x, y) \) is reducible then the corresponding \( p(c, y) \) is also reducible. Hence if \( p(c, y) \) is irreducible for some constant \( x = c \) it implies that \( p(x, y) \) is irreducible.

---

of 5 points, symmetric about a line \( z = c \), for then we know that the solution lies somewhere on \( z = c \). Then for a set of \( n \) points distributed equally and symmetrically about our chosen axis \( z = c \), (when \( n \) is odd, 1 point lies on this axis), problem (1) is transformed to the following problem.

\[
\text{minimize } y \quad p(y) = \sum_{i=1}^{n} \sqrt{(x - \alpha_i)^2 + (y - b_i)^2} \quad (5)
\]

We choose a configuration of 5 points symmetric about the line \( z = 0 \). One of the points \( p \) lies on the line and has coordinates \((0, c)\) on the \( z = 0 \) axis. The value of \( c \) changes the configuration of points in that for \( c = 5, 1 \) and \( 4 \) we have respectively 3, 4 or 5 points on the convex hull of our set of 5 points. (Figure 4.4).

---

Let \((a_1, b_1) = (3, 0), (a_2, b_2) = (1, 3), (a_3, b_3) = (0, c), (a_4, b_4) = (-1, 3)\) and \((a_5, b_5) = (-3, 0)\).
Lemma (7.1). The polynomial $p(x)$ is reducible over $\mathbb{Q}$. The degree of the extension $\mathbb{Q}[x]/(p(x))$. 

Lemma (7.2). Let $\mathbb{K}$ be an extension field of $\mathbb{Q}$. If $p(x)$ is reducible over $\mathbb{K}$, then there exists an irreducible factor of $p(x)$ over $\mathbb{K}$. 

We need to find the solutions of $\mathbb{Q}(\sqrt{3}, \sqrt{5})/\mathbb{Q}$ which is also the solution for the polynomial $p(x) = x^4 + 1$. 

We use this to find the extension $\mathbb{Q}[x]/(p(x))$. 

The polynomial $p(x)$ is irreducible over $\mathbb{Q}$. 

We need to find the solution $\mathbb{Q}(\sqrt{3}, \sqrt{5})/\mathbb{Q}$.
we denote $[F:Q] =$ degree of $F$ over $Q$, (the dimension of $F$ as a vector space over $Q$).

Consider all the points $(x,y)$ in the real Euclidean plane, both of whose coordinates $x$, $y$ are in $Q$. This set of points is called the plane of $Q$. A point is constructible from $Q$ iff we can find a finite number of real numbers $\alpha_1, \ldots, \alpha_n$ such that (i) $[Q(\alpha_1):Q] = 1$ or $2$ and (ii) $[Q(\alpha_1, \ldots, \alpha_i):Q(\alpha_1, \ldots, \alpha_{i-1}]] = 1$ or $2$, and such that our point lies in the plane of $Q(\alpha_1, \ldots, \alpha_n)$. It follows that if $\alpha$ is constructible then $\alpha$ lies in some extension of $Q$, of degree a power of $2$.

We know that a real number $\alpha$ is algebraic over $Q$ iff $Q(\alpha)$ is a finite extension of $Q$. Further $\alpha$ is said to be algebraic of degree $n$ over $Q$ if it satisfies a non-zero polynomial of degree $n$ but no non-zero polynomial of lower degree. Also if $\alpha$ is algebraic of degree $n$ over $Q$, then $[Q(\alpha):Q] = n$. This together with our discussion of constructibility above gives the following important criterion for non-constructibility.

Lemma 4.2: [He75] If the real number $\alpha$ satisfies an irreducible polynomial over $Q$ of degree $n$ and if $n$ is not a power of $2$, then $\alpha$ is not constructible.

If $p(y) \in Q[y]$, a finite extension $E$ of $Q$ is said to be a splitting field over $Q$ for $p(y)$ if $E$ contains all the roots of $p(y)$ and $E$ is a minimal field extension of $Q$. Alternatively, $E$ is a splitting field of $p(y)$ over $Q$ if $E$ is a minimal extension of $Q$ in which $p(y)$ has $n$ roots, where $n =$ degree of $p(y)$.

Given a polynomial $p(y)$ in $Q[y]$, the polynomial ring in $y$ over $Q$, we shall associate with $p(y)$ a group, $G = \text{Gal}(p(y))$, the Galois group of $p(y)$. The Galois group turns out to be a certain permutation group of the roots of the polynomial. It is actually defined as a certain group of automorphisms of the splitting field of $p(y)$ over $Q$. A beautiful duality, expressed in the fundamental theorem of Galois Theory exists between the subgroups of the Galois group and the subfields of the splitting field. From this one can derive a condition for the solvability by means of radicals of the roots of a polynomial in terms of the algebraic structure of its Galois group. As a special case we can give a criterion for non-constructibility by straight-edge and compass constructions, which follows from the above criterion.

Lemma 4.3: If $E$ is the splitting field over $Q$ for an irreducible polynomial $p(y)$, and if the order of its galois group, $\sigma[\text{Gal}(p(y))] = [E:Q]$, is not a power of $2$, then the roots of $p(y)$ are not constructible.

A few additional theorems from Galois theory of use to us here are.

Lemma 4.4: [Ca71] For a finite field $F$, $\text{dim}_F p(y)$ factors over $F$ into $k$ different irreducible factors, $p(y) = q_1(y) \cdots q_k(y)$, where degree $q_i(y) = n_i$. Then $\text{Gal}(p(y))$ is cyclic and is generated by a permutation containing $k$ cycles with orders $n_1, \ldots, n_k$.

The shape of a permutation of degree $n$ is the partition of $n$ induced by the lengths of the disjoint cycles of the permutation. The factorization of a polynomial modulo any prime $p$ also induces a partition, namely the partition of the degree of $p(y)$ formed by the degree of the factors. The above Lemma states that the degree partition of the factors of $p(y)$ modulo $p$ is the shape of the generating permutation of the group, $\text{Gal}(p(y))$, which is furthermore cyclic.

Lemma 4.5: [Ca71] Let $p(y) \in Z[y]$ have roots $\alpha_1, \ldots, \alpha_n$ and let $p^*(y) \in Z_p[y]$, be the polynomial $p(y)$ mod $p$. If the roots of $p^*(y)$ are $\alpha_1^*, \ldots, \alpha_n^*$, then the
Usually the decision that \( \text{Gal}(p(y)) = S_n \) is reached even after much less than \( n+1 \) trials as a consequence of the evolving pattern of permutations occurring in \( \text{Gal}(p(y)) \) and the application of known theorems of permutation groups.

We are now ready to prove our main theorem, but first let us indulge in some definitions. A polynomial \( p(y) \in Q[y] \) is called solvable over \( Q \) if there is a finite sequence of fields \( Q = F_0 < F_1 < \cdots < F_k \), where \( F_i < F_{i+1} \) implies that \( F_i \) is a subfield of \( F_{i+1} \) and a finite sequence of integers \( n_0, \ldots, n_k \) such that \( F_{i+1} = F_i(a_i) \) with \( a_i \in F_i \) and if all the roots of \( p(y) \) lie in \( F_k \), that is, \( E \subseteq F_k \), where \( E \) is the splitting field of \( p(y) \). \( F_k \) is called a radical extension of \( Q \). Furthermore, we know from Galois theory that

**Lemma 4.9**: \( p(y) \in Q[y] \) is solvable by radicals over \( Q \) if the Galois group over \( Q \) of \( p(y) \), \( \text{Gal}(p(y)) \) is a solvable group.

**Lemma 4.10**: \( p(y) \in Q[y] \) is solvable by radicals over \( Q \) if the Galois group over \( Q \) of \( p(y) \), \( \text{Gal}(p(y)) \) is not solvable for \( n \geq 5 \).

**Theorem 4.11**: The Fermat-Weber problem, in general, is not solvable by radicals over \( Q \) for \( n \geq 5 \).

**Proof**: Restating the assertion, we need to show that the polynomial \( p(y) \) of Table 1 is not solvable by radicals over \( Q \). We note from Table 1 that for the 'good' primes \( p = 19, 31, \) and \( 37 \), the degrees of the irreducible factors of \( p(y) \mod p \) gives us a 2 + 5 permutation, an 8 cycle and a 7 cycle, which is enough to establish, from Lemma 4.8 that \( \text{Gal}(p(y)) = S_8 \), the symmetric group of degree 8. Lemma 4.10 tells us that this is not a solvable group and hence our assertion follows from Lemma 4.9.

### 4.3. The Line-restricted Weber problem

We now consider the line-restricted Weber problem.

Given \( n \) fixed destination points as before in the plane with coordinates \( (x_i, y_i) \), we need to determine the location \((x, y)\) of a single source point, restricted to lie on certain given line, such that the sum of the Euclidean distances from this source to each of the destinations is minimized.

We consider two different positions (and orientations) of this line, since the algebraic degree of the solution point varies with the relative positions of the line and the fixed destination points. For the non-trivial case of 3 destination points consider the solution restricted to a line passing through one of the points and either not intersecting the convex-hull (of the destination points), or passing through the convex-hull, figure 4.5.

**Lemma 4.12**: For the above cases of figure 4.5, the minimal polynomial \( p(y) \) (Table 2) of degree 8 is irreducible over \( Q \). Furthermore, this polynomial is not solvable by radicals over \( Q \).
Theorem 1.6: The line intersected more problems. In general, it is not solvable by end.

Proof: From Lemma 1.6.

Let $A$ be a point in a non-solvable group. Hence, our assertion follows. In fact, the line intersected more problems. In general, it is not solvable by end.

From Lemma 1.6, we obtain that $A$ is a non-solvable group. Hence, our assertion follows. In fact, the line intersected more problems. In general, it is not solvable by end.

Proof: Since $A$ is a non-solvable group, a good prime $p$ follows that $A$ is a non-solvable group. Hence, our assertion follows. In fact, the line intersected more problems. In general, it is not solvable by end.

Figure 1.6: For the above cases of Figure 1, the minimal polynomial $P(x)$ is given by

$$P(x) = x^2 + x + 1,$$
Table 3

\[
\begin{align*}
\text{minimize } & \quad f(y)=\sqrt{(y^3)^2+4} + \sqrt{(y^3)^2+1} + \sqrt{y^2+1} \\
\text{df/}dy & = \frac{3(3y^2)}{\sqrt{(y^3)^2+4} + (y^3)/\sqrt{(y^3)^2+1} + y/\sqrt{y^2+1}} = 0 \\
Q : p(y) & = 3y^{12} + 72y^{11} + 780y^{10} + 4992y^9 + 20772y^8 + 58500y^7 + 113010y^6 \\
& + 155449y^5 + 156012y^4 + 119040y^3 + 51870y^2 + 972y^1 + 729 = 0 \\
\text{deg}(p(y)) & = 2^{9}\cdot 3^{5}\cdot 13^4 \\
\text{Mod 7} : p(y) & = (y^2 + 3y^11 + y^10 + 2y^9 + y^8 + 2y^7 + 2y^6 + 3y^5 + 2y^4 + 2y^3 + 2y^2 + 2y + 2) = 0 \\
\text{Mod 10} : p(y) & = (y^6 + 8y^5 + 7y^4 + 7y^3 + 9y^2 + 5y^1) = 0 \\
\text{Mod 61} : p(y) & = (y^6 + 13y^5 + 3y^4) \\
& + (y^3 + 27y^2 + 19y^1 + 19y^0) + (y^6 + y^4 + 10y^3 + 25y^2 + 21y + 23) = 0 \\
\end{align*}
\]


We mention in passing that for the case of the line passing through 2 of the 3 given destination points, the solution to the Line restricted Weber problem coincides with the projection of the 3rd point onto that line and so is constructible.

Furthermore, the case of \( n=5 \) for the symmetric Fermat-Weber problem is equivalent to the (weighted) case, \( n=3 \), of the Line-restricted Weber problem, where the line is the axis of symmetry, which passes through one of the destination points. (And hence the algebraic degree of the solutions is the same). On the other hand the above case of \( n=3 \) of the Line-restricted Weber problem where the line does not pass through any of the destination points is equivalent to the case of \( n=6 \) for the symmetric Fermat-Weber problem, (the line becoming the axis of symmetry as before). The solutions of these cases are as expected, of higher algebraic degree.

Conclusions

Restating the open problem of [LM83], the best thing one would like is an efficient way of computing the Galois group of a polynomial. Having obtained the Galois group one can show various wonderful properties of the geometric optimization problem on hand. Notably, if the polynomial is solvable one can obtain the tower decomposition of the solvable Galois group and thereby obtain the various extension fields of the rationals, where the roots of the polynomial lie. Calculating the \( k \)th order radical in which the solution can be expressed is another possibility.

N.B.: In producing the examples of this paper we used the computer algebra system Maple, (actually Vaxima on Unix).

Acknowledgements: Thanks to John Hopcroft for his early inspiration which led to this paper.


