Available online at www.sciencedirect.com
SCIENCE DIRECT®
Journal of Complexity 22 (2006) 250–274
Journal of COMPLEXITY
www.elsevier.com/locate/jco

Computational complexity of diagram satisfaction in Euclidean geometry

Nathaniel Miller*

Department of Mathematical Sciences, University of Northern Colorado, Ross Hall, Greeley, CO 80639, USA

Received 2 March 2005; accepted 2 September 2005
Available online 21 November 2005

Abstract

In this paper, it is shown that the problem of deciding whether or not a geometric diagram in Euclidean Geometry is satisfiable is NP-hard and in PSPACE, and in fact has the same complexity as the satisfaction problem for a fragment of the existential theory of the real numbers. The related problem of finding all of the possible (satisfiable) diagrams that can result when a segment of a diagram is extended is also shown to be NP-hard.

Keywords: Euclidean geometry; Diagrams; Computational complexity

1. Introduction

Case analysis has long been a sticking point in attempts to understand how diagrams are used in mathematics, particularly in geometry. When using diagrams, a question that immediately arises is: how many different diagrams must I consider? Indeed, one of the earliest criticisms of Euclid's Elements, which argues extensively using diagrams, was that he did not distinguish enough cases. Some more recent commentators have argued that proofs that rely on diagrams are inherently informal because the process of finding all of the cases that need to be considered is a non-algorithmic human process that is perhaps even open-ended: each time someone finds a case that has not been dealt with yet, a new proof has to be constructed for that case. However, this is incorrect. In other work (see [3] or [4]), it is shown how to construct a formal system FG for Euclidean Geometry that uses geometric diagrams as its syntactic objects and is similar enough to the way that people normally use diagrams informally in geometry to directly formalize the

* Fax: +1 970 351 1225.
E-mail address: nat@alumni.princeton.edu.

0885-064X/$ - see front matter © 2005 Elsevier Inc. All rights reserved.
doi:10.1016/j.jco.2005.09.003

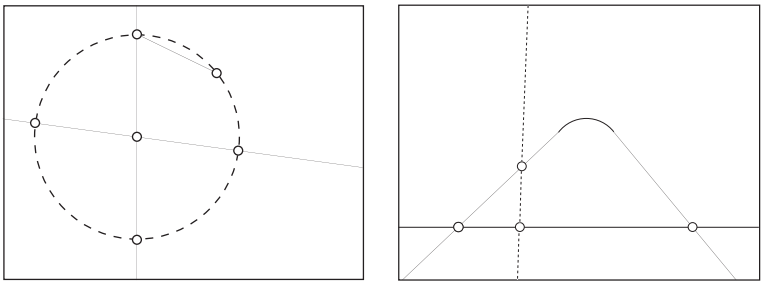

Fig. 1. Two primitive diagrams.

kinds of proofs that Euclid gives in Book I of *The Elements*. In this paper, it is shown how we can use a well-defined syntax and semantics of diagrams to understand how case analysis works in Euclidean geometry, and that this problem is NP-hard.

In fact, case analysis in Euclidean geometry can be done by an algorithm, which has been implemented in the computerized formal proof system CDEG (Computerized Diagrammatic Euclidean Geometry). CDEG automatically does this case analysis in the course of constructing a proof. However, there is a sense in which geometric case analysis is difficult. Consider the problem of finding all of the new diagrams that can result from extending a line segment in a given diagram outward until it intersects another element of the diagram. The algorithm used by CDEG solves this problem, but sometimes returns extra diagrams that do not represent any physically realizable situation. There is in fact a computable algorithm which returns precisely those diagrams that represent the physical situations that could occur when you extend the line—that is, it does not return any extra unrealizable cases—but we will show that the problem of finding precisely the realizable cases is at least NP-hard, and in fact has the same complexity as the satisfaction problem for a particular fragment of the existential theory of real arithmetic.

2. Basic syntax and semantics of diagrams

Because the purpose of this paper is not to explain the formal system FG for doing Euclidean Geometry, just barely enough will be explained here for the complexity discussion to make sense. The interested reader is referred to [3] or [4] for more details.

If we want to discuss the role of diagrams in geometry, we must first say what is meant by the term diagram in this context. Fig. 1 shows two examples of the sort of diagrams we want to consider. They contain dots and edges representing points, straight lines and circles in the plane, but note that a diagram may not look exactly like the configuration of lines and circles that it represents; in fact, it may represent an impossible configuration, like the second diagram in Fig. 1.

It is possible to give a formal syntax for such diagrams that eliminates many, though not all, of the unsatisfiable diagrams. The formal definition of what constitutes a diagram and of the formal deductive system FG that manipulates them is discussed at great length in [3,4]; the details will not be given here. The general idea is that a diagram is a kind of planar graph. The formal definition as a kind of graph with lots of conditions, however, is just meant to mimic the ways that diagrams are used informally. An important idea here is that two diagrams represent the same situation if

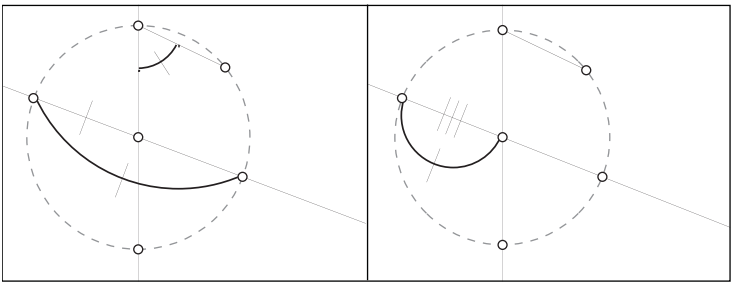

Fig. 2. A diagram array containing two marked versions of the first primitive diagram in Fig. 1.

they are topologically equivalent as graphs in the plane; in this case, we say that the diagrams are equivalent.

In order to distinguish the pieces of diagrams from the geometric objects that they represent, we will refer to the diagrammatic objects that represent points as *dots*; the diagrammatic objects that represent lines as *dlines*; the diagrammatic objects that represent circles as *dcircs*; the diagrammatic objects that represent angles as *di-angles*; and the outer box surrounding a diagram as the *frame*. We call a dline that intersects the frame twice and therefore represents an infinite line a *proper dline*, and we call one that does not intersect the frame at all and therefore represents a line segment a *dseg*. We allow dsegs and di-angles in diagrams to be marked with slash marks to show that they represent congruent pieces, as is traditional in Euclidean geometry. In this case, we will refer to a set of slash marks as a *marker*. (Di-angles will be marked by drawing a bold arc through the angle and then putting the marker on the arc.) We also allow several diagrams to be joined together along their frames to represent the existence of multiple possible situations; we call the resulting object a diagram array. Diagram arrays are allowed to be empty. Fig. 2 shows a diagram array containing two different marked versions of the first diagram in Fig. 1.

We would like to discuss the relationship between diagrams and real geometric figures. By a *Euclidean plane*, we mean a plane together with a finite number of designated points, circles, rays, lines, and line segments, such that all the points of intersection of the designated circles, rays, etc. are included among the designated points. The elements of Euclidean planes are the objects about which we would like to be able to reason. We consider the designated points of a Euclidean plane to divide its circles and lines into pieces, which we call *designated edges*.

It is very easy to turn a Euclidean plane P into a diagram. We can do this as follows: pick any new point n in P, pick a point p_l on each designated line l of P, and let m be the maximum distance from n to any designated point, any p_l, or to any point on a designated circle. m must be finite, since P only contains a finite number of designated points, lines and circles. Let R be a circle with center n and radius of length greater than m, and let F be a rectangle lying outside of R. Then if we let D be a diagram whose frame is F, whose segments are the parts of the edges of P that lie inside F, whose dots are the designated points of P, and whose dlines and dcircles are the connected components of the lines and circles of P, then D is a diagram that we call P's *canonical* (*unmarked*) *diagram*. (Strictly speaking, we should say a canonical diagram, since the diagram we get depends on how we pick n and the p_l; but all the diagrams we can get are equivalent.) We can also find P's *canonical marked diagram* by marking as equal those dsegs or di-angles in D that correspond to congruent segments or angles in P. These canonical diagrams

give us a convenient way of identifying which Euclidean planes are represented by a given diagram.

Definition 1. A Euclidean plane M is a *model* of the primitive diagram D (in symbols, $M \models D$, also read as "M satisfies D") if

(1) M's canonical unmarked diagram is equivalent to D's underlying unmarked diagram, and
(2) if two segments or di-angles are marked equal in D, then the corresponding segments or di-angles are marked equal in M's canonical marked diagram.

M is a model of a diagram array if it is a model of any of its component diagrams.

This definition just says that $M \models D$ if M and D have the same topology and any segments or angles that are marked congruent in D really are congruent in M. Note that this definition makes a diagram array into a kind of disjunction of its primitive diagrams and that the empty diagram array therefore has no models.

It is immediate from the definitions that every Euclidean plane is the model of some diagram, namely its canonical underlying diagram, and that if D and E are equivalent diagrams, then $M \models D$ iff $M \models E$. In other words, the satisfaction relation is well defined on equivalence classes of diagrams. The full converse of this statement, that if $M \models D$ and $M \models E$, then $D \equiv E$, is not true, since D and E may have different markings. However, it is true if D and E are unmarked.

3. Construction rules

The formal system FG also has rules to manipulate and reason with diagrams: rules of construction, inference, and transformation. These rules allow FG to mimic traditional informal proofs in geometry. Of these, the ones that are relevant to the case analysis problem are the construction rules. The rules work as follows: each rule, when applied to a given diagram D yields a diagram array of all the diagrams that satisfy the rule. Two of these rules (there are several others) are given in Table 1. These two rules correspond directly to Euclid's first two postulates, which state that a line segment can be drawn between any two points, and that any line segment can be extended indefinitely.

As a relatively simple example of how these rules work, consider the diagram shown in Fig. 3. What happens if we apply rule C1 to this diagram in order to connect points C and D? We get the diagram array of all diagrams extending the given diagram in which there is a dseg connecting points C and D. In this case, there are nine different topologically distinct possibilities, as CDEG confirms, which are shown in Fig. 4. (In order to check carefully that these are the only possibilities, you need to know the precise definition of what constitutes a legitimate diagram; the interested reader is again referred to [3] or [4].)

Table 1
Some of the diagram construction rules

Diagram construction rules
C1. If there is not already one existing, a dseg may be added whose endpoints are any two given existing distinct dots.
C2. Any dseg can be extended to a proper dline.

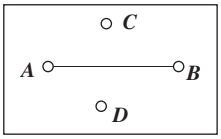

Fig. 3. What can happen when points C and D are connected?

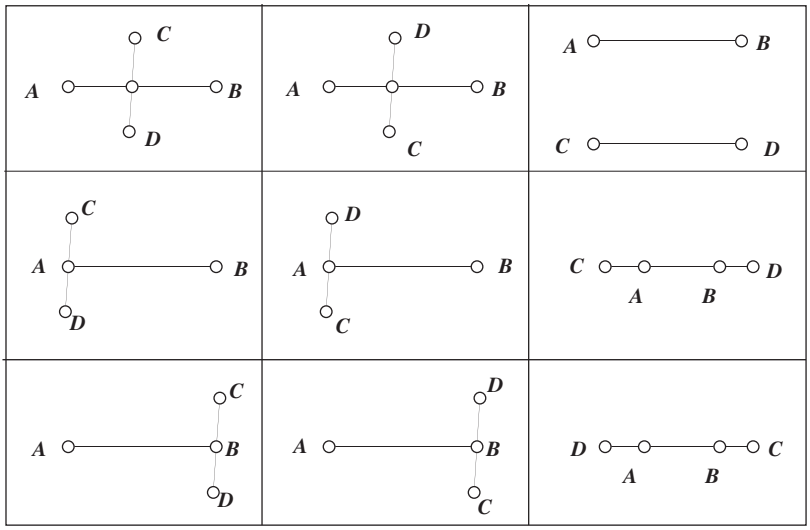

Fig. 4. The result of applying rule C1 to points C and D in the diagram in Fig. 3.

4. Satisfiable and unsatisfiable diagrams

In the preceding sections, we gave a thumbnail sketch of how we can define a formal system in which geometric diagrams are defined as a type of planar graphs that meet certain conditions. We also gave a definition of what it means for a Euclidean plane to satisfy a diagram, and pointed out that the definition of a diagram was designed to eliminate as many unsatisfiable diagrams as possible.

An obvious question, then, is how well could our definitions have succeeded at eliminating these unrealizable situations? That is: did we succeed in eliminating *all* of the unsatisfiable diagrams, or are there still some diagrams with no models?

The answer is that there are indeed unsatisfiable diagrams. Fig. 5 shows a diagram which is unsatisfiable because according to Desargues' theorem, in any model of this diagram, line XY would have to intersect line $B'C'$ at point Z. The usual proof of Desargues' theorem is as follows: imagine that the diagram shows a two-dimensional projection of a three-dimensional picture of a pyramid with base ABC and summit vertex E. Then the triangles ABC and $A'B'C'$ determine two different planes P_1 and P_2 in three-space. Lines AB and $A'B'$ meet in three-space, because they both lie in the plane determined by triangle ABE, and since AB lies in P_1 and $A'B'$ lies in P_2, their point of intersection X must lie in the intersection of P_1 and P_2. Likewise, if Y is the intersection of AC and $A'C'$ and Z is the intersection of BC and $B'C'$, then Y and Z also lie in the intersection of the two planes. Since two planes intersect in a line, this means that X, Y, and Z

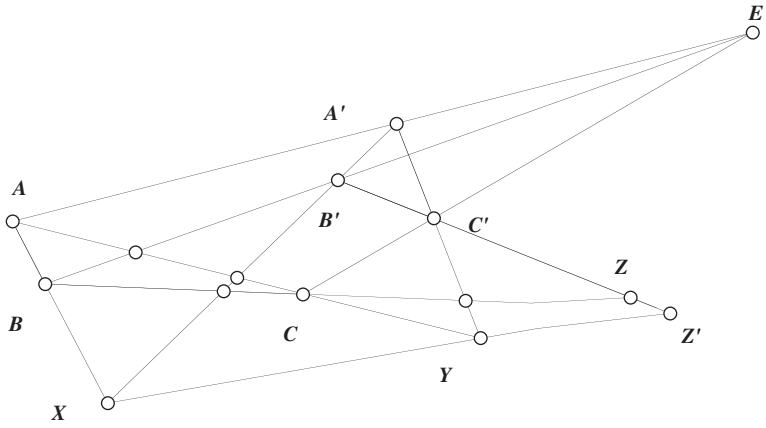

Fig. 5. An unsatisfiable diagram.

should be collinear; but in the given diagram, point Z does not fall on the line XY, so the diagram is unsatisfiable. This example is particularly striking because it contains nothing but unmarked dsegs, but there are many other possible examples.

This shows that our definition of what constitutes a diagram is still too broad in one sense, because there are diagrams that are unsatisfiable. However, it is hard to imagine a reasonable definition of what a diagram is under which the diagram in Fig. 5 does not qualify. Our next question, then, is: is there some additional set of conditions that could be added to eliminate all of the unsatisfiable diagrams? It would be extremely convenient to find such a set of conditions. For example, consider what happens when we apply one of the construction rules to a satisfiable diagram. We get back an array of possible results. As it now stands, we know that because the construction rules are sound, at least one of the diagrams that we get back must be satisfiable, but many of them may not be satisfiable. If we could find a set of conditions that eliminated these unsatisfiable diagrams, then we would not have to waste our time looking at these extra cases. So such a set of conditions would be extremely powerful.

The very fact that such a set of conditions would be so powerful might make us suspect they would be *too* powerful, and that such a set of conditions is impossible to find. But somewhat surprisingly, it can in fact be computed whether or not a given diagram is satisfiable. Our definition of satisfiability can be translated into the first-order language of real arithmetic, and we can apply Tarski's theorem, which says that there is a procedure for deciding if a given sentence of the first-order language of arithmetic is true or false (as a statement about the real numbers). (See [7].) In fact, in Section 6, we show the formula that translates our definition of diagram satisfaction is Σ_1, which means that the decision procedure is in PSPACE. This means that we could define a diagram to be *strongly well-formed* if it is a diagram under our old definition, and Tarski's decision procedure says that it is satisfiable. Then the strongly well-formed diagrams would exactly capture the possible configurations of real Euclidean planes. The problem with this approach is that the decision procedure given by Tarski's theorem can take intractably long to run. A set of conditions that correctly determine if a diagram is satisfiable but take exponentially long to evaluate are not really useful.

So a new question is: is there a procedure that determines whether or not a given diagram is satisfiable in a reasonable amount of time? To be more specific, our question becomes: is there a

polynomial-time algorithm for determining whether or not a given diagram is satisfiable? It turns out that the answer to this question is no, assuming that P $\neq$ NP. In Section 5 we will show that the diagram satisfiability problem is NP-hard. Thus, we will have shown that diagram satisfaction is NP-hard and in PSPACE.

5. NP-hardness

In this section, we show that the problem of determining if a given diagram is satisfiable is NP-hard. The problem of determining if a given boolean formula is satisfiable is well known to be NP-complete, so it suffices to show how to reduce the boolean satisfiability problem to the diagram satisfiability problem in log-space. (See [2] for a proof that the boolean satisfiability problem is NP-complete.)

We will consider here only boolean formulas without OR gates; as is well known, a boolean formula that includes OR gates can be converted into one without OR gates by using De Morgan's Laws to rewrite the formula $(A \vee B)$ as $\neg(\neg A \wedge \neg B)$. Thus, we will consider a boolean formula to be a string composed of three types of symbols: boolean variables $(x_1, x_2, x_3, \ldots)$, parenthesis, and the logical operators AND $(\wedge)$ and NOT $(\neg)$. A string of these symbols is a boolean formula if it is a boolean variable, in which case it is called an atomic formula, or if it is of the form $(A \wedge B)$ or $\neg A$, where A and B are boolean formulas. The *proper subformulas* of a formula are defined as follows: the proper subformulas of $A \wedge B$ are A, B, and the proper subformulas of A and B; the proper subformulas of $\neg A$ are A and its proper subformulas; and atomic formulas have no proper subformulas. The *subformulas* of F are F along with all of the proper subformulas of F. Given an assignment of truth values (true and false) to the boolean variables of a formula, the truth value of the formula can be determined as follows: if the formula is a boolean variable, then the truth value of the formula is the same as the truth value of the variable; if the formula is of the form $(A \wedge B)$, then the formula is true if and only if both of the subformulas A and B are true; and if the formula is of the form $\neg A$ then it is true just if A is false. A boolean formula is satisfiable if and only if there is an assignment of truth values to its propositional variables that makes the whole formula true.

For every boolean formula F, we can define a corresponding diagram $D(F)$ which is satisfiable if and only if F is. The basic idea of the construction is to come up with a piece of a diagram that can be satisfied in exactly two ways. One of these ways will stand for the truth value TRUE, and the other will stand for the truth value FALSE. For this purpose, we will use the diagram shown in Fig. 7, which can be satisfied in exactly two different ways, as represented in the left half of Fig. 6. (In one of these ways, $a_i = Y_1$; in the other, $a_i = Y_2$.) We will have one such diagrammatic piece for each boolean variable, and then we will find appropriate ways to combine the pieces to mimic the logical connectives AND and NOT. Thus, we will be able to come up with a diagram that is satisfied by a sequence of triangles just if our formula is satisfied by the corresponding sequence of truth values.

We define $D(F)$ formally as follows: let $F_1, F_2, F_3, \ldots, F_{f-1}$ be the proper subformulas of F, arranged in order of increasing complexity, so that if F_j is a subformula of F_k then $j \leqslant k$, and let F_f be F. For each i, $0 \leqslant i \leqslant f + 1$, we are going to define a subdiagram $D_i(F)$. $D(F)$ will be a diagram that contains disjoint copies of all of these subdiagrams. In order to construct $D(F)$, we first pick $6f + 8$ distinct markers: six di-angle markers a_i, b_i, g_i, e_i, h_i, and m_i for each subformula F_i of F; six other di-angle markers, labeled here by one slash mark and the names Y_1, Y_2, Z, H, and R; and two dseg markers, shown as two and three slash marks. Marker R will be used to mark right angles and will also be designated by drawing the usual right angle

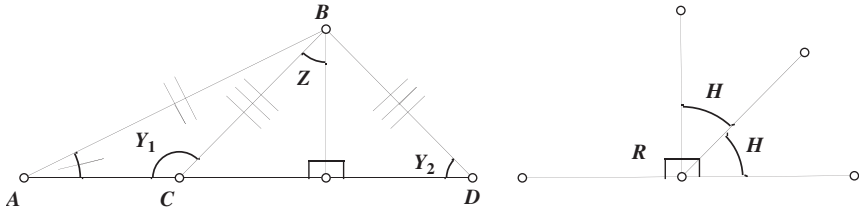

Fig. 6. $D_0(F)$.

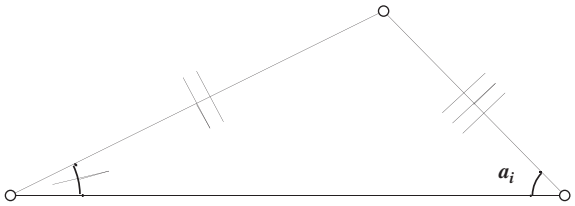

Fig. 7. Subdiagram contained in $D_i(F)$ if F_i is an atomic formula or a conjunction.

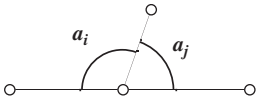

Fig. 8. $D_i(F)$ when F_i is $\neg F_j$.

symbol in the diagrams. In the following discussion, the marker names will also be used to refer to the measures of the angles that they represent; it should be clear from context which meaning is intended.

We let $D_0(F)$ be the subdiagram shown in Fig. 6 and let $D_{f+1}(F)$ be the subdiagram shown in Fig. 10. For $1 \leqslant i \leqslant f$, we define $D_i(F)$ as follows:

- If F_i is atomic, then $D_i(F)$ is the subdiagram shown in Fig. 7.
- If F_i is $\neg F_j$, then $D_i(F)$ is the subdiagram shown in Fig. 8.
- If F_i is $(F_j \wedge F_n)$, then $D_i(F)$ is the subdiagram that contains both of the subdiagrams shown in Figs. 7 and 9.

Let $D_i^\circ(F)$ be the (smallest) diagram containing $D_0, D_1, \ldots, D_i$ as disjoint subdiagrams, and let $D(F) = D_{f+1}^\circ(F)$.

Note that if F has length n, then it has at most n subformulas, and the subdiagram put into $D(F)$ for each subformula has a size bounded by a constant, so the size of $D(F)$ is linear in the length of F. In fact, in order to compute $D(F)$ from F, the only thing that you need to keep track of is which subformula in F you are currently on, which you can do in log-space.

Now, note that Fig. 7 along with Fig. 6 forces angle a_i to be equal to one of Y_1 or Y_2, since there are only two triangles that can be built with the given angle and sides. (For proof, see [1, pp. 304–307].) It follows by induction on the complexity of F_i that a_i must be equal either to Y_1 or else to Y_2, for every i: if F_i is atomic or a conjunction, then $D_i(F)$ contains Fig. 7, and if it is $\neg F_j$, then a_i is supplementary to a_j, which is one of Y_1 or Y_2 by the inductive hypothesis,

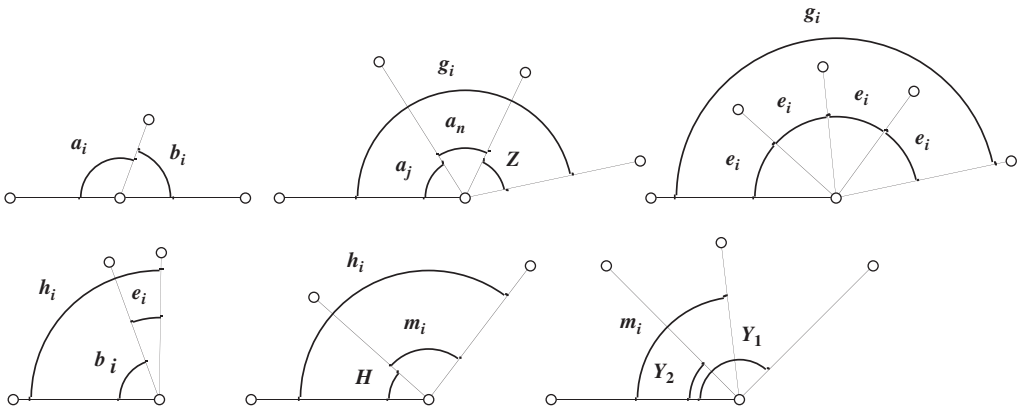

Fig. 9. Subdiagram contained in $D_i(F)$ if F_i is $(F_j \wedge F_n)$.

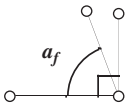

Fig. 10. $D_{f+1}(F)$.

and so a_i is Y_2 or Y_1, since Y_1 and Y_2 are supplements. Furthermore, note that Y_2 and Z are complementary, so $Y_2 = 90° - Z$ and $Y_1 = 90° + Z$.

We want to show these two possible values of each a_i correspond to the two possible truth values of F_i, so that $a_i < 90°$ iff F_i is true. More specifically, we will say that a model M *agrees* with assignment t on a subformula F_i if $a_i = Y_2$ and F_i is true under t, or if $a_i = Y_1$ and F_i is false under t. We will show that any model of any of the D_i° that agrees with a truth assignment t on the atomic subformulas of F also agrees with t on all of the other subformulas. This means that the subdiagrams in Figs. 8 and 9 act as logical NOT and AND gates, so that if F_i is $(F_j \wedge F_n)$, then $a_i < 90°$ iff $a_j < 90°$ and $a_n < 90°$, and if F_i is $\neg F_j$, then $a_i < 90°$ iff $a_j \geqslant 90°$. This is shown in the following lemma:

Lemma 2. *Let t be a function assigning truth values to the boolean variables of F. Then*:

(a) *For each $i \leqslant f$, there is a model M_i of $D_i^\circ(F)$ that agrees with t on all atomic subformulas F_k with $k \leqslant i$.*
(b) *Furthermore, if M is any model of $D_i^\circ(F)$ that agrees with t on all atomic subformulas F_k with $k \leqslant i$, then it must also agree with t on all other subformulas F_k with $k \leqslant i$.*

Proof. By induction on i.

Base case: $i = 0$. In this case, we just have to show that $D_0^\circ(F)$, which is just $D_0(F)$, has a model. It has one: take any isosceles triangle, draw a perpendicular through the vertex, extend the base to one side, and connect it to the vertex to get a model of the first half of D_0. To get a model of the other half, bisect a straight angle into two right angles, then divide one of the right angles into two equal pieces.

Inductive cases:

(1) F_i is an atomic formula of the form x_j. By the inductive hypothesis, $D^\circ_{i-1}(F)$ has a model M_{i-1} that agrees with t on all subformulas F_k such that $k < i$. That model satisfies all of $D^\circ_i(F)$ except for the triangle added by $D_i(F)$. Any triangle added by $D_i(F)$ will have to be congruent to one of the two triangles ABC or ABD; conversely, any model that extends M_{i-1} and contains a new disjoint triangle which is congruent to ABC or ABD will satisfy $D^\circ_i(F)$. So if $t(x_j) =$ true, let M_i be such an extension of M_{i-1} in which the new triangle is congruent to ABC, and otherwise let it be such an extension of M_{i-1} in which the new triangle is congruent to ABD. Then M_i agrees with t on F_i, which shows part (a). For part (b), note that any model M of $D_i(F)$ that agrees with t on atomic formulas has a submodel that is a model of D_{i-1}; so by part (b) of the inductive hypothesis, $a_k = Y_2$ iff F_k is true under t when $k \leqslant i - 1$, and this is also true when $k = i$, since F_i is atomic. This shows part (b).

(2) F_i is a formula of the form $\neg F_j$. By the inductive hypothesis, there is a model M_{i-1} of $D^\circ_{i-1}(F)$ that agrees with t on atomic formulas (by part (a)), and therefore agrees with t on all subformulas F_i with $k < i$ (by part (b)). Let M_i be a model extending M_{i-1} that also contains a new straight angle divided into two pieces such that the clockwise piece is congruent to the other angles that are marked by a_j. Then M_i is a model of $D_i(F)$, proving part a. To show part (b), note that any model M of $D_i(F)$ that agrees with t on the atomic formulas of F must agree with t on all the subformulas F_k such that $k < i$, as before, so it suffices to show that it agrees with t on F_i, that is, that $a_i = Y_2$ in M iff F_i is true under t. Since j must be less than i (because the subformulas of F were arranged in order of increasing complexity), and a_i and a_j are supplements, $a_i = Y_2$ iff $a_j = Y_1$ iff F_j is false under t iff F_i is true under t. This proves (b).

(3) F_i is a formula of the form $(F_j \wedge F_n)$. We want to show that $D_i(F)$ forces a_i to be less than $90°$ iff a_j and a_n are both less than $90°$. First note that in any model of $D_i(F)$, $m_i = (a_j + a_n + Z)/4 + b_i - 45°$ (from pieces 2–5 of Fig. 9); $Y_1 < m_i < Y_2$ (from piece 6 of Fig. 9); and b_i is equal to either Y_1 or Y_2 ($90° + Z$ or $90° - Z$), because it is supplementary to a_i. There are three cases to consider:

   (a) $a_j = a_n = Y_2 = 90° - Z$. Then $m_i = (180° - Z)/4 + b_i - 45° = 45° - Z/4 + b_i - 45° = b_i - Z/4$. Since $90° - Z < m_i < 90° + Z$, this means that $90° - Z < b_i - Z/4 < 90° + Z$, so $90° - 3Z/4 < b_i < 90° + 5Z/4$. Since b_i is either $90° - Z$ or $90° + Z$, this means that it must be $90 + Z = Y_1$. So since a_i is supplementary to b_i, this means that $a_i = Y_2$.

   (b) $a_j = Y_1$ and $a_n = Y_2$, or $a_j = Y_2$ and $a_n = Y_1$. Then $m_i = b_i + Z/4$; so $90° - 5Z/4 < b_i < 90° + 3Z/4$ so b_i must be equal to Y_2 to keep m_i between Y_1 and Y_2; so $a_i = Y_1$.

   (c) $a_j = a_n = Y_1 = 90° + Z$. Then $m_i = b_i + 3Z/4$, and b_i must be equal to Y_2 as before, so $a_i = Y_1$.

   To prove part (b), let M be any model of $D^\circ_i(F)$ that agrees with t on atomic formulas. By the inductive hypothesis, M agrees with t on all subformulas F_k with $k < i$. It suffices to show that M agrees with t on F_i. Now, if F_i is true under t, then F_j and F_n must also be true under t (by the truth table for AND), so by the inductive hypothesis, $a_j = a_n = Y_2$, which is the first case above, so $a_i = Y_2$ in M, as required. On the other hand, if F_i is false under t, then one of F_j or F_n must also be false. So, by the inductive hypothesis, one or both of a_i and a_n must be equal to Y_2. So we are in either the second or third case above, and in both of these cases, $a_i = Y_1$ in M, also as required. This proves part (b). For part (a), note that we can extend M_{i-1} with pieces satisfying $D_i(F)$ as long as we make a_i equal to Y_1 or Y_2 according to the

three cases above, because the first five pieces of Fig. 9 serve only to define m_i, and the last piece will be satisfied as long as that m_i lies between Y_2 and Y_1.

This proves the lemma. □

We can now prove the following theorem:

Theorem 3. *Any boolean formula F is satisfiable if and only if* $D(F)$ *is satisfiable.*

Proof. ($\Rightarrow$) Assume that F is satisfiable. Then there is a truth assignment t of the boolean variables of F under with F is true. So, by the lemma, there is a model M of $D^\circ_f(F)$ that agrees with t on F_i for all $i \leqslant f$. In particular, M agrees with t on F_f. Since $F_f = F$ and F is true under t, this means that $a_f = Y_2$ in M. The only difference between $D^\circ_f(F)$ and $D(F)$ is that $D(F)$ contains $D_{f+1}(F)$. So it suffices to show that there is an extension of M that satisfies $D_{f+1}(F)$. But $D_{f+1}(F)$ just says that $a_f < 90°$. So since $a_f = Y_2 < 90°$ in M, M can be extended to a model M' that satisfies $D(F)$.

($\Leftarrow$) Now assume that F is unsatisfiable. Then F is false under all possible truth assignments of its boolean variables. So, by the lemma, $a_f = Y_1 > 90°$ in any model of $D^\circ_f(F)$, because any model of $D^\circ_f(F)$ agrees with some possible truth assignment on atomic formulae. So no model of $D^\circ_f(F)$ can be extended to a model of $D(F)$, since a_f would have to be less than 90° in any such model because it would have to satisfy D_{f+1}. So $D(F)$ is unsatisfiable. □

We have shown how to reduce the question of whether or not a given boolean formula is satisfiable to the question of whether or not a given diagram is satisfiable, and this reduction can be done in log-space. Therefore, because the boolean satisfiability problem is NP-complete, we have the following corollary:

Corollary 4. *The diagram satisfiability problem is NP-hard under log-space computable many-one reductions.*

Next, consider the *case analysis problem*: let D be a satisfiable primitive diagram, and let $S(D)$ be the set of satisfiable diagrams that can result from extending a given line segment in D outward until it intersects another segment. The problem of figuring out exactly what diagrams are in $S(D)$ is also NP-hard. To see this, let F be a boolean formula, let $D''_{f+1}(F)$ be the subdiagram shown in Fig. 11, and let $D''(F)$ be the smallest diagram containing $D_0(F), D_1(F), \ldots, D_f(F)$ and $D''_{f+1}(F)$. Then F is satisfiable iff $D''(F)$ has a model in which $a_f = Y_2$, iff $D''(F)$ has a

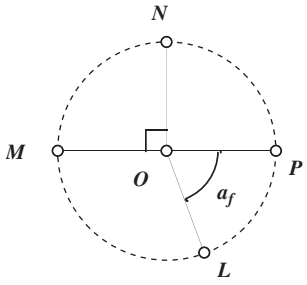

Fig. 11. $D''_{f+1}(F)$.

satisfiable extension in which dseg OL is extended into di-angle MON. So since $D''(F)$ can also be produced from F in log-space, we also have:

Corollary 5. *The case analysis problem is also NP-hard under log-space many-one reductions.*

6. Defining diagram satisfaction in first-order logic

In this section we show how to define diagram satisfaction in the theory of real arithmetic. Given a diagram D that contains dots $d_1, \ldots, d_n$, dlines $l_1, \ldots, l_p$, and dcircles $c_1, \ldots, c_k$, such that all of D's dlines are proper, such that there are at least two dots on every dline and dcircle in D, and such that there is at least one dot on every ray coming out of each marked angle in D, we will define a formula of the language of real arithmetic

$$\mathrm{CF}_D(x_1, y_1, \ldots, x_n, y_n, m_1, b_1, \ldots, m_p, b_p, c_{x1}, c_{y1}, r_1, \ldots, c_{xk}, c_{yk}, r_k)$$

which is satisfiable over the real numbers iff D is satisfiable. This formula will be called D's *corresponding formula*. It will be constructed so that $x_1, \ldots, r_k$ will satisfy CF iff the Euclidean plane containing the designated points $(x_1, y_1), \ldots, (x_n, y_n)$, the lines satisfying the equations $y = m_1x+b_1, \ldots, y = m_px+b_p$, and the circles satisfying the equations $(x-c_{x1})^2+(y-c_{y1})^2 = r_1^2, \ldots, (x-c_{xk})^2 + (y-c_{yk})^2 = r_k^2$ satisfies D.

We will build up this formula from many simpler formulas. First of all, we are going to want the points (x_1, y_1) to be distinct and the r_i to be positive, so we define formulas that say this:

$$\mathrm{DISTINCT}(x_1, \ldots, r_k) := \textstyle\bigwedge_{i\neq j}((x_i \neq x_j) \vee (y_i \neq y_j)),$$
$$\mathrm{POSR}(r_1, \ldots, r_k) := \textstyle\bigwedge_{i=1}^{k} r_i > 0.$$

We also need to make sure that every pair of circles and/or lines intersects only at designated points. In order to do this, we must first define several other predicates.

$$\mathrm{LC1INT}(m, b, c_x, c_y, r) := r^2(m^2+1) = m^2c_x^2 - 2mc_yc_x + 2mbc_x - 2bc_y + c_y^2 + b^2,$$

$$\mathrm{LC2INT}(m, b, c_x, c_y, r) := r^2(m^2+1) > m^2c_x^2 - 2mc_yc_x + 2mbc_x - 2bc_y + c_y^2 + b^2.$$

LC1INT and LC2INT hold if the given line intersect the given circle exactly once or twice, respectively. This is because a formula from analytic geometry tells us that the distance between the point (x_0, y_0) and the line $y = mx + b$ is given by

$$\frac{|x_0 - y_0 + b|}{\sqrt{m^2+1}}.$$

So LC1INT says that the distance between the given line and the center of the given circle is equal to the radius of the circle, while LC2INT says that it is less than the radius of the circle.

$$\begin{aligned}\mathrm{CC1INT}&(c_{x1}, c_{y1}, r_1, c_{x2}, c_{y2}, r_2) :=\\ &[(c_{x1}-c_{x2})^2 + (c_{y1}-c_{y2})^2 = r_1^2 + r_2^2 + 2r_1r_2] \vee\\ &[(c_{x1}-c_{x2})^2 + (c_{y1}-c_{y2})^2 = r_1^2 + r_2^2 - 2r_1r_2],\end{aligned}$$

$$\begin{aligned}\mathrm{CC2INT}&(c_{x1}, c_{y1}, r_1, c_{x2}, c_{y2}, r_2) :=\\ &[(c_{x1}-c_{x2})^2 + (c_{y1}-c_{y2})^2 > r_1^2 + r_2^2 + 2r_1r_2] \wedge\\ &[(c_{x1}-c_{x2})^2 + (c_{y1}-c_{y2})^2 < r_1^2 + r_2^2 - 2r_1r_2].\end{aligned}$$

CC1INT and CC2INT hold if two given circles intersect once or twice, respectively. CC1INT says that the distance between the centers of the given circles is equal to either the sum or the difference of their radii, while CC2INT says that it is between the sum and the difference of the radii.

We are now in a position to define the predicates that say that any intersections of lines and/or circles occur only at one of the designated points.

$$\begin{aligned}&\mathrm{LLINT}(m_{j_1}, b_{j_1}, m_{j_2}, b_{j_2}) :=\\ &\quad (m_{j_1} = m_{j_2}) \vee\\ &\quad \textstyle\bigvee_{i=1}^{n}((m_{j_1}x_i + b_{j_1} = y_i) \wedge (m_{j_2}x_i + b_{j_2} = y_i)).\end{aligned}$$

LLINT says that if the two given lines are not parallel, then one of the given points lies on both of them, *i.e.*, at their point of intersection.

$$\begin{aligned}\mathrm{LCINT}(m, b, c_x, c_y, r) := (&\mathrm{LC1INT}(m, b, c_x, c_y, r)\\ &\to \textstyle\bigvee_{i=1}^{n}((mx_i + b = y_i) \wedge ((x_i - c_x)^2 + (y_i - c_y)^2 = r)))\\ &\quad \wedge (\mathrm{LC2INT}(m, b, c_x, c_y, r)\\ &\to \textstyle\bigvee_{i\neq j}((mx_i + b = y_i) \wedge ((x_i - c_x)^2 + (y_i - c_y)^2 = r)\\ &\quad \wedge (mx_j + b = y_j) \wedge ((x_j - c_x)^2 + (y_j - c_y)^2 = r))),\end{aligned}$$

$$\begin{aligned}&\mathrm{CCINT}(c_{x1}, c_{y1}, r_1, c_{x2}, c_{y2}, r_2)\\ &\quad := (\mathrm{CC1INT}(c_{x1}, c_{y1}, r_1, c_{x2}, c_{y2}, r_2) \to \textstyle\bigvee_{i=1}^{n}((x_i - c_{x1})^2 + (y_i - c_{y1})^2 = r_1)\\ &\qquad \wedge((x_i - c_{x2})^2 + (y_i - c_{y2})^2 = r_2))) \wedge (\mathrm{CC2INT}(c_{x1}, c_{y1}, r_1, c_{x2}, c_{y2}, r_2)\\ &\quad \to \textstyle\bigvee_{i\neq j}((x_i - c_{x1})^2 + (y_i - c_{y1})^2 = r_1) \wedge ((x_i - c_{x2})^2 + (y_i - c_{y2})^2 = r_2)\\ &\qquad \wedge((x_j - c_{x1})^2 + (y_j - c_{y1})^2 = r_1) \wedge ((x_j - c_{x2})^2 + (y_j - c_{y2})^2 = r_2))).\end{aligned}$$

LCINT says that if the line and circle intersect once, then one of the given points lies on both of them, and if they intersect twice, then two of the listed points lie on both of them; and CCINT says the same thing for two circles. We can now define the tuple $(x_1, \ldots, r_k)$ to be *well-formed* if its points are distinct, its r's are positive, and all its points of intersection between circles and lines are given by listed points; that is, if it satisfies

$$\begin{aligned}\mathrm{WF}(x_1, \ldots, r_k) :=\ &\mathrm{DISTINCT}(x_1, \ldots, r_k) \wedge \mathrm{POSR}(x_1, \ldots, r_k)\\ &\wedge \textstyle\bigwedge_{i\neq j}\mathrm{LLINT}(m_i, b_i, m_j, b_j)\\ &\wedge \textstyle\bigwedge_{i,j}\mathrm{LCINT}(m_i, b_i, c_{xj}, c_{yj}, r_j)\\ &\wedge \textstyle\bigwedge_{i\neq j}\mathrm{CCINT}(c_{xi}, c_{yi}, r_i, c_{xj}, c_{yj}, r_j).\end{aligned}$$

Next, we want to say that the given points, lines, and circles have the right graph structure. First, we define predicates that say that a point lies on a line, on a circle, or at the center of a circle.

$$\begin{aligned}&\mathrm{ONLINE}(x, y, m, b) := y = mx + b,\\ &\mathrm{ONCIRC}(x, y, c_x, c_y, r) := (x - c_x)^2 + (y - c_y)^2 = r^2,\\ &\mathrm{ISCENT}(x, y, c_x, c_y, r) := (x = c_x) \wedge (y = c_y).\end{aligned}$$

We now want to make sure that the points occur in the right order. For this purpose, we use two predicates that say that two points are adjacent to one another on a given line and that one point follows another in the clockwise direction on a given circle:

$$\begin{aligned}
&\mathrm{ADJONLINE}(x_{j_1}, y_{j_1}, x_{j_2}, y_{j_2}, m, b) :=\\
&\quad \mathrm{ONLINE}(x_{j_1}, y_{j_1}, m, b) \wedge \mathrm{ONLINE}(x_{j_2}, y_{j_2}, m, b)\\
&\qquad \wedge \bigwedge_{i \neq j_1, j_2} (\mathrm{ONLINE}(x_i, y_i, m, b)\\
&\qquad\qquad \rightarrow (\neg((x_{j_1} < x_i < x_{j_2}) \vee (x_{j_2} < x_i < x_{j_1}))))
\end{aligned}$$

and

$$\begin{aligned}
&\mathrm{CADJONCIRC}(x_{j_1}, y_{j_1}, x_{j_2}, y_{j_2}, c_x, c_y, r) :=\\
&\quad \mathrm{ONCIRC}(x_{j_1}, y_{j_1}, c_x, c_y, r) \wedge \mathrm{ONCIRC}(x_{j_2}, y_{j_2}, c_x, c_y, r)\\
&\quad \wedge [(y_{j_1} \geqslant c_y \wedge y_{j_2} \geqslant c_y \wedge x_{j_1} < x_{j_2}) \rightarrow\\
&\qquad \bigwedge_{i \neq j_1, j_2} ((\mathrm{ONCIRC}(x_i, y_i, c_x, c_y, r) \wedge y_i \geqslant c_y)\\
&\qquad\qquad \rightarrow (\neg((x_{j_1} < x_i < x_{j_2}))))]\\
&\quad \wedge [(y_{j_1} \geqslant c_y \wedge y_{j_2} \geqslant c_y \wedge x_{j_2} < x_{j_1}) \rightarrow\\
&\qquad \bigwedge_{i \neq j_1, j_2} (((\mathrm{ONCIRC}(x_i, y_i, c_x, c_y, r)\\
&\qquad\qquad \rightarrow ((x_{j_1} < x_i < x_{j_2}) \wedge y_1 > c_y))))]\\
&\quad \wedge [(y_{j_1} \leqslant c_y \wedge y_{j_2} \leqslant c_y \wedge x_{j_2} < x_{j_1}) \rightarrow\\
&\qquad \bigwedge_{i \neq j_1, j_2} ((\mathrm{ONCIRC}(x_i, y_i, c_x, c_y, r) \wedge y_i \leqslant c_y)\\
&\qquad\qquad \rightarrow (\neg((x_{j_2} < x_i < x_{j_1}))))]\\
&\quad \wedge [(y_{j_1} \leqslant c_y \wedge y_{j_2} \leqslant c_y \wedge x_{j_2} > x_{j_1}) \rightarrow\\
&\qquad \bigwedge_{i \neq j_1, j_2} (((\mathrm{ONCIRC}(x_i, y_i, c_x, c_y, r)\\
&\qquad\qquad \rightarrow ((x_{j_1} < x_i < x_{j_2}) \wedge y_1 < c_y))))]\\
&\quad \wedge [(y_{j_1} \leqslant c_y \wedge y_{j_2} \leqslant c_y \wedge x_{j_2} < x_{j_1}) \rightarrow\\
&\qquad \bigwedge_{i \neq j_1, j_2} ((\mathrm{ONCIRC}(x_i, y_i, c_x, c_y, r) \wedge y_i \leqslant c_y)\\
&\qquad\qquad \rightarrow (\neg((x_{j_2} < x_i < x_{j_1}))))]\\
&\quad \wedge [(y_{j_1} \leqslant c_y \wedge y_{j_2} \leqslant c_y \wedge x_{j_2} > x_{j_1}) \rightarrow\\
&\qquad \bigwedge_{i \neq j_1, j_2} (((\mathrm{ONCIRC}(x_i, y_i, c_x, c_y, r)\\
&\qquad\qquad \rightarrow ((x_{j_1} < x_i < x_{j_2}) \wedge y_1 < c_y))))]\\
&\quad \wedge [(y_{j_1} < c_y \wedge y_{j_2} > c_y) \rightarrow\\
&\qquad \bigwedge_{i \neq j_1, j_2} (\mathrm{ONCIRC}(x_i, y_i, c_x, c_y, r)\\
&\qquad\qquad \rightarrow (\neg(x_i < x_{j_1} \wedge y_i < c_y) \wedge \neg(x_i < x_{j_2} \wedge j_i > c_y)))]\\
&\quad \wedge [(y_{j_1} > c_y \wedge y_{j_2} < c_y) \rightarrow\\
&\qquad \bigwedge_{i \neq j_1, j_2} (\mathrm{ONCIRC}(x_i, y_i, c_x, c_y, r)\\
&\qquad\qquad \rightarrow (\neg(x_i > x_{j_1} \wedge y_i > c_y) \wedge \neg(x_i > x_{j_2} \wedge j_i < c_y)))].
\end{aligned}$$

Now we are in a position to write down a formula saying that the points, lines, and circles have the right graph structure. (Actually, it says something slightly stronger, since any points on circles must lie in the right orientation.) First we define the sets

$$\mathrm{ADJ}_x = \{j_1, j_2 | d_{j_1}, d_{j_2} \text{ are adjacent on } x\},$$

where x can either be one of the lines l_i or one of the circles c_i. Then we can write the desired formula as follows:

$$\begin{aligned}
&\mathrm{GF}_D(x_1, \ldots, r_k) := \\
&\quad [\textstyle\bigwedge_{i=1}^{p} \bigwedge_{\{j|d_j \text{ lies on } l_i\}} \mathrm{ONLINE}(x_j, y_j, m_j, b_j))] \\
&\quad \wedge [\textstyle\bigwedge_{i=1}^{p} \bigwedge_{\{j|d_j \text{ does not lie on } l_i\}} \neg\mathrm{ONLINE}(x_j, y_j, m_j, b_j)] \\
&\quad \wedge [\textstyle\bigwedge_{i=1}^{k} \bigwedge_{\{j|d_j \text{ lies on } c_i\}} \mathrm{ONCIRC}(x_j, y_j, c_{xj}, c_{yj}, r_j))] \\
&\quad \wedge [\textstyle\bigwedge_{i=1}^{k} \bigwedge_{\{j|d_j \text{ does not lie on } c_i\}} \neg\mathrm{ONCIRC}(x_j, y_j, c_{xj}, c_{yj}, r_j)] \\
&\quad \wedge [\textstyle\bigwedge_{i=1}^{k} \bigwedge_{\{j|d_j \text{ is the center of } c_i\}} \mathrm{ISCENT}(x_j, y_j, c_{xj}, c_{yj}, r_j))] \\
&\quad \wedge [\textstyle\bigwedge_{i=1}^{k} \bigwedge_{\{j|d_j \text{ is not the center of } c_i\}} \neg\mathrm{ISCENT}(x_j, y_j, c_{xj}, c_{yj}, r_j)] \\
&\quad \wedge [\textstyle\bigwedge_{i=1}^{p} \bigwedge_{(j_1,j_2)\in \mathrm{ADJ}_{l_i}} \mathrm{ADJONLINE}(x_{j_1}, y_{j_1}, x_{j_2}, y_{j_2}, m_j, b_j))] \\
&\quad \wedge [\textstyle\bigwedge_{i=1}^{p} \bigwedge_{(j_1,j_2)\notin \mathrm{ADJ}_{l_i}} \neg\mathrm{ADJONLINE}(x_{j_1}, y_{j_1}, x_{j_2}, y_{j_2}, m_i, b_i)))] \\
&\quad \wedge [\textstyle\bigwedge_{i=1}^{p} \bigwedge_{(j_1,j_2)\in \mathrm{ADJ}_{c_i}} \mathrm{ADJONCIRC}(x_{j_1}, y_{j_1}, x_{j_2}, y_{j_2}, c_{xi}, c_{yi}, r_i))] \\
&\quad \wedge [\textstyle\bigwedge_{i=1}^{p} \bigwedge_{(j_1,j_2)\notin \mathrm{ADJ}_{c_i}} \neg\mathrm{ADJONCIRC}(x_{j_1}, y_{j_1}, x_{j_2}, y_{j_2}, c_{xi}, c_{yi}, r_i))].
\end{aligned}$$

We also need to make sure that all the points lie in the correct regions. To do this, we use the following formulas:

$$\begin{aligned}
&\mathrm{INCIRC}(x, y, c_x, c_y, r) := (x - c_x)^2 + (y - c_y)^2 < r^2, \\
&\mathrm{OUTCIRC}(x, y, c_x, c_y, r) := (x - c_x)^2 + (y - c_y)^2 > r^2, \\
&\mathrm{CW}(l_{x1}, l_{y1}, l_{x2}, l_{y2}, x, y) := (l_{x2} - l_{x1})(y - l_{y1}) < (x - l_{x1})(l_{y2} - l_{y1}), \\
&\mathrm{CCW}(l_{x1}, l_{y1}, l_{x2}, l_{y2}, x, y) := (l_{x2} - l_{x1})(y - l_{y1}) > (x - l_{x1})(l_{y2} - l_{y1}).
\end{aligned}$$

INCIRC and OUTCIR say that the given point is inside or outside of the given circle. CW and CCW say that the point (x, y) lies on the clockwise or counterclockwise side of the directed line from (l_{x1}, l_{y1}) to (l_{x2}, l_{y2}). This meaning of the formulas follows from the geometric meaning of the cross product. CW says that the z component of the cross product of the vector from (l_{x1}, l_{y1}) to (l_{x2}, l_{y2}) and the vector from (l_{x1}, l_{y1}) to (x, y) is negative, and CCW says that it is positive.

We can now define a formula that says that all of the points lie in the correct regions of the diagram. Recall that we require D to have at least two different dots on each dline l_i, so we can pick $a_{i,1}$ and $a_{i,2}$ so that $d_{a_{i,1}}$ and $d_{a_{i,2}}$ both lie on l_i and they are not equal. Define

$$\begin{aligned}
\mathrm{CWP}(i) &:= \{j|d_j \text{ lies on the clockwise side of the directed line from } d_{a_{i,1}} \text{ to } d_{a_{i,2}}\}; \\
\mathrm{CCWP}(i) &:= \{j|d_j \text{ lies on the counterclockwise side of} \\
&\qquad \text{the directed line from } d_{a_{i,1}} \text{ to } d_{a_{i,2}}\}; \\
\mathrm{INP}(i) &:= \{j|d_j \text{ lies inside } c_i\}; \quad \text{and} \\
\mathrm{OUTP}(i) &:= \{j|d_j \text{ lies outside } c_i\}.
\end{aligned}$$

We can now define the formula as follows:

$$\begin{aligned}
\mathrm{CREG}_D(x_1, \ldots, r_k) := &\textstyle\bigwedge_{i=1}^{p} \bigwedge_{j\in \mathrm{CWP}(i)} \mathrm{CW}(x_{a_{i,1}}, y_{a_{i,1}}, x_{a_{i,2}}, y_{a_{i,2}}, x_j, y_j) \wedge \\
&\textstyle\bigwedge_{i=1}^{p} \bigwedge_{j\in \mathrm{CCWP}(i)} \mathrm{CCW}(x_{a_{i,1}}, y_{a_{i,1}}, x_{a_{i,2}}, y_{a_{i,2}}, x_j, y_j) \wedge \\
&\textstyle\bigwedge_{i=1}^{k} \bigwedge_{j\in \mathrm{INP}(i)} \mathrm{INCIRC}(x_j, j_j, c_{xj}, c_{yj}, r_j) \wedge \\
&\textstyle\bigwedge_{i=1}^{k} \bigwedge_{j\in \mathrm{OUTP}(i)} \mathrm{OUTCIRC}(x_j, j_j, c_{xj}, c_{yj}, r_j).
\end{aligned}$$

The last thing that we need to do is to make sure that segments and angles that are marked congruent are really the same size. To do this, we need a predicate that says that the segment between (x_{i_1}, y_{i_1}) and (x_{i_2}, y_{i_2}) is congruent to the segment between (x_{i_3}, y_{i_3}) and (x_{i_4}, y_{i_4}):

$$\begin{aligned}&\mathrm{CONGS}_D(x_{i_1}, y_{i_1}, x_{i_2}, y_{i_2}, x_{i_3}, y_{i_3}, x_{i_4}, y_{i_4}) :=\\ &((x_{i_2} - x_{i_1})^2 + (y_{i_2} - y_{i_1})^2 = ((x_{i_4} - x_{i_3})^2 + (y_{i_4} - y_{i_3})^2.\end{aligned}$$

We also need a similar predicate that says that the angle θ_1 determined by the three points (x_{i_1}, y_{i_1}), (x_{i_2}, y_{i_2}), and (x_{i_3}, y_{i_3}) is congruent to the angle θ_1 determined by the points (x_{i_4}, y_{i_4}), (x_{i_5}, y_{i_5}), and (x_{i_6}, y_{i_6}). (Here, the first point gives the vertex of the angle, and the sides of the angle are rays going through the other two points, which are given in clockwise order.) First we use the fact that

$$\frac{\mathbf{A} \cdot \mathbf{B}}{|\mathbf{A}||\mathbf{B}|} = \cos\theta$$

for any two vectors $\mathbf{A}$ and $\mathbf{B}$ and angle θ between them to write a formula that says that $\cos^2\theta_1 = \cos^2\theta_2$, as follows:

$$\begin{aligned}&\mathrm{ECOS2}_D(x_{i_1}, y_{i_1}, x_{i_2}, y_{i_2}, x_{i_3}, y_{i_3}, x_{i_4}, y_{i_4}, x_{i_5}, y_{i_5}, x_{i_6}, y_{i_6}) :=\\ &\quad((x_{i_3} - x_{i_1})(x_{i_2} - x_{i_1}) + (y_{i_3} - y_{i_1})(y_{i_2} - y_{i_1}))^2\\ &\quad\times((x_{i_5} - x_{i_4})^2 + (y_{i_5} - y_{i_4})^2)((x_{i_6} - x_{i_4})^2 + (y_{i_6} - y_{i_4})^2) =\\ &\quad((x_{i_6} - x_{i_4})(x_{i_5} - x_{i_4}) + (y_{i_6} - y_{i_4})(y_{i_5} - y_{i_4}))^2\\ &\quad\times((x_{i_2} - x_{i_1})^2 + (y_{i_2} - y_{i_1})^2)((x_{i_3} - x_{i_1})^2 + (y_{i_3} - y_{i_1})^2).\end{aligned}$$

If the squares of the two cosines are equal, then the cosines are equal as long as they have the same sign; we can check this by making sure that the two dot products have the same sign:

$$\begin{aligned}&\mathrm{ECOS}_D(x_{i_1}, y_{i_1}, x_{i_2}, y_{i_2}, x_{i_3}, y_{i_3}, x_{i_4}, y_{i_4}, x_{i_5}, y_{i_5}, x_{i_6}, y_{i_6}) :=\\ &\quad\mathrm{ECOS2}_D(x_{i_1}, y_{i_1}, x_{i_2}, y_{i_2}, x_{i_3}, y_{i_3}, x_{i_4}, y_{i_4}, x_{i_5}, y_{i_5}, x_{i_6}, y_{i_6}) \wedge\\ &\quad((x_{i_3} - x_{i_1})(x_{i_2} - x_{i_1}) + (y_{i_3} - y_{i_1})(y_{i_2} - y_{i_1}))\\ &\quad\times((x_{i_6} - x_{i_4})(x_{i_5} - x_{i_4}) + (y_{i_6} - y_{i_4})(y_{i_5} - y_{i_4})) > 0.\end{aligned}$$

Now, if the cosines of the two angles are equal, then either the angles are equal, or else they sum to 360°. So we can check that they are the same by making sure that they are both greater than or both less than 180°. We can do this by making sure that (x_{i_3}, y_{i_3}) falls on the same (clockwise or counterclockwise) side with respect to the vector from (x_{i_1}, y_{i_1}) to (x_{i_2}, y_{i_2}) as (x_{i_6}, y_{i_6}) falls with respect to the vector from (x_{i_4}, y_{i_4}) to (x_{i_5}, y_{i_5}). The following predicate accomplishes this:

$$\begin{aligned}&\mathrm{CONGANG}_D(x_{i_1}, y_{i_1}, x_{i_2}, y_{i_2}, x_{i_3}, y_{i_3}, x_{i_4}, y_{i_4}, x_{i_5}, y_{i_5}, x_{i_6}, y_{i_6}) :=\\ &\quad\mathrm{ECOS}_D(x_{i_1}, y_{i_1}, x_{i_2}, y_{i_2}, x_{i_3}, y_{i_3}, x_{i_4}, y_{i_4}, x_{i_5}, y_{i_5}, x_{i_6}, y_{i_6}) \wedge\\ &\quad(\mathrm{CW}(x_{i_1}, y_{i_1}, x_{i_2}, y_{i_2}, x_{i_2}, y_{i_2}) \to \mathrm{CW}(x_{i_4}, y_{i_4}, x_{i_5}, y_{i_5}, x_{i_6}, y_{i_6})).\end{aligned}$$

We are now in a position to define a formula that says that all of the angles and segments marked congruent are congruent. First we need to define the following sets:

$$\begin{aligned}\text{Cong-segs}_D := \{&(d_{i_1}, d_{i_2}, d_{i_3}, d_{i_4}) | \text{the dseg through } d_{i_1} \text{ and } d_{i_2}\\ &\text{is marked congruent to the dseg through } d_{i_3} \text{ and } d_{i_4}\},\\ \text{Cong-angs}_D := \{&(d_{i_1}, d_{i_2}, d_{i_3}, d_{i_4} d_{i_5}, d_{i_6}) | \text{the di-angle given by } d_{i_1}, d_{i_2},\\ &\text{and } d_{i_3} \text{ is marked congruent to the di-angle given by } d_{i_4}, d_{i_5}, \text{ and } d_{i_6}\}.\end{aligned}$$

We can use these to define the desired formula:

$$\begin{aligned}&\mathrm{CONG}_D(x_1,\ldots,r_k) :=\\ &\quad[\textstyle\bigwedge_{(d_{i_1},d_{i_2},d_{i_3},d_{i_4})\in \text{Cong-segs}_D}\mathrm{CONGS}_D(x_{i_1},y_{i_1},\ldots,x_{i_4},y_{i_4})]\wedge\\ &\quad[\textstyle\bigwedge_{(d_{i_1},\ldots,d_{i_6})\in\text{Cong-angs}_D}\mathrm{CONGANG}_D(x_{i_1},y_{i_1},\ldots,x_{i_6},y_{i_6})].\end{aligned}$$

Finally, we can define the originally promised formula CF as follows:

$$\begin{aligned}\mathrm{CF}_D(x_1,\ldots,r_k) :=\ &\mathrm{WF}(x_1,\ldots,r_k)\wedge\mathrm{GF}_D(x_1,\ldots,r_k)\\ &\wedge\mathrm{CREG}_D(x_1,\ldots,r_k)\wedge\mathrm{CONG}_D(x_1,\ldots,r_k).\end{aligned}$$

Note the following facts:

(1) $\mathrm{CF}_D(x_1,\ldots,r_k)$ is quantifier free.
(2) D is a satisfiable diagram iff the existential closure of CF_D is satisfiable over the Real numbers, which is a decidable question by Tarski's theorem. (In fact, the sentence in question contains only existential quantifiers, and it is known that such sentences can be decided in polynomial space. For details, see [5].)
(3) If A is a diagram array consisting of primitive diagrams $\{D_1,\ldots,D_m\}$, and each of these primitive diagrams contains only proper dlines, has at least two dots on each dline and dcircle, and has at least one dot on every ray coming out of a marked angle, then A is satisfiable iff the existential closure of $\mathrm{CF}_{D_1}\vee\cdots\vee\mathrm{CF}_{D_m}$ is satisfiable over the Reals.
(4) Given any diagram array E, we can use the construction rules to find a diagram array E' whose primitive diagrams only contain proper dlines and have at least two dots on each dline and dcircle and one dot on every ray that emanates from a marked angle, such that E' is satisfiable iff E is.

It follows from these facts that the general question of satisfiability of diagrams is decidable.

7. Using diagrams to encode real arithmetic

In the preceding two sections, we have shown that the diagram satisfiability problem is NP-hard and in PSPACE. A natural direction for further inquiry, then, is to see if we can improve either of these bounds. In particular, if we could improve the second bound to show that the diagram satisfaction problem was in NP, then it would be NP-complete. While this is still an open possibility, it seems unlikely. In this section, we will show how to encode a significant fragment of the existential theory of the real numbers in our diagrams. We will then use this encoding to show that diagram satisfaction has the same complexity as the satisfiability problem for this fragment, which we will call PER-DNF. Thus, the diagram satisfaction problem is in NP if and only if the satisfiability problem for PER-DNF is also in NP. While this is conceivable, it seems unlikely. Any sentence in the existential theory of the reals can be translated into a sentence in PER-DNF, and the existential theory of the reals is only known to be in PSPACE; all known algorithms for deciding this theory are exponential in the number of variables that can occur in a sentence. (See [5] for details.) On the other hand, the translation from a sentence of the existential theory of the reals into a sentence of PER-DNF can lead to an exponential blow up of the size of the sentence, so it is still possible that this fragment has lower complexity than the full existential theory of the reals. However, the fragment is already strong enough to include formulas that express systems of equalities and inequalities of polynomials whose degree can be exponential in the length of the formula.

The fragment of the existential theory of the reals that we want to consider is the fragment that only contains positively bounded existential quantifiers, and whose formulas are in disjunctive normal form with positive literals of the form $a = b$ and $a < b$. We will call this fragment PER-DNF, for "the Positive Existential theory of Real arithmetical sentences in Disjunctive Normal Form." We will define PER-DNF precisely as follows:

(1) PER-DNF contains a set of *numerals*. 1 is a numeral, and if n is a numeral, then $S(n)$ is also a numeral. These numerals are intended to represent the positive integers, and the length of a numeral is directly proportional to the integer it represents.
(2) The *atomic terms* of PER-DNF are the numerals along with the variable symbols $x_1, x_2, x_3, \ldots$.
(3) The *terms* of PER-DNF are defined as follows: if a and b are atomic terms, then a, b, $a + b$, and $a \times b$ are terms. Note that this definition *is not* recursive—each term contains at most one operation symbol.
(4) The *atomic formulae* of PER-DNF are the following: if s and t are terms, then $s = t$ and $s < t$ are atomic formulae.
(5) The *conjunctions* of PER-DNF are defined in terms of the atomic formulae: all atomic formulae are conjunctions, and if A and B are conjunctions, then $(A \wedge B)$ is a conjunction.
(6) The *disjunctions* of PER-DNF are defined as follows: all conjunctions are disjunctions, and if C and D are disjunctions, then $(C \vee D)$ is also a disjunction.
(7) Finally, we define the formulas of PER-DNF as follows: if $F(x_{i_1}, \ldots, x_{i_k})$ is a disjunction containing k different variable symbols, then

$$(\exists x_{i_1} > 0) \cdots (\exists x_{i_k} > 0)(F(x_{i_1}, \ldots, x_{i_k}))$$

is a formula of PER-DNF.

Expressions in PER-DNF may be quite awkward and long, since the numerals are long, each term contains at most one operation symbol, and negation is not available. However, PER-DNF is still powerful enough to express any sentence that is expressible in the full existential theory of the reals with positive existential quantifiers and positive rational constants. We will call this theory the positive existential theory of the reals. In order to see that we can rewrite any sentence in this theory as a sentence in PER-DNF, first note that we can express a positive rational constant a/b by the expression $(\exists x_i > 0)(b \times x_i = a)$; we can represent expressions involving polynomials as conjunctions of simpler expressions; and subtraction and division can be rewritten in terms of addition and multiplication. For example, the expression $y = ax^2 - b$ can be rewritten as

$$(\exists x_2 > 0)((x_2 = x \times x) \wedge (x_2 \times a = y + b)).$$

So if we start with any formula in the full positive existential theory of the reals, we can use these tricks to rewrite it as a formula only containing atomic formulae from PER-DNF. Then, to turn it into a formula of PER-DNF, we just have to put it into disjunctive normal form with positive literals. We can do this in the usual way, using the distributive law and De Morgan's laws. First, we can move all negations inwards using De Morgan's laws and then remove pairs of negations, leaving us with a formula in which all negations occur as part of a literal (an atomic formula or its negation). Next, we can rewrite any negated atomic formulae as disjunctions of un-negated atomic formulae. That is, we can rewrite $\neg(a = b)$ as $(a < b \vee b < a)$, and we can rewrite $\neg(a < b)$ as $(a = b \vee b < a)$. Note that everything that we have done so far has at most linearly increased the size of our formula. Finally, we can use the distributive law of AND over OR to

move all ANDs inward, leaving us with a formula in disjunctive normal form, which is therefore in PER-DNF. Unfortunately, however, this last step can lead to an exponential blowup in the size of our formula; thus, it is still possible that PER-DNF has lower complexity than the full positive existential theory of the reals.

We can take this idea one step further and encode the full existential theory of the reals by encoding an arbitrary real number r as a pair of positive real numbers (s, a), where s is equal to either 1 or 2 depending on whether r is negative or positive and $a = |r|$, except when $r = 0$; in that case, we let s equal 3 and let a equal anything we like. We can then use this convention to rewrite our original formula with only positive quantifiers. For example, we would rewrite the formula

$$(\exists r_1)(\exists r_2)(\exists r_3)(r_1 \times r_2 = r_3)$$

as

$$\begin{aligned}(\exists s_1)(\exists a_1)(\exists s_2)(\exists a_2)(\exists s_3)(\exists a_3)((a_1 \times a_2 = a_3) \wedge (((s_1 = 1 \wedge s_2 = 1) \wedge s_3 = 2)\\ \vee((s_1 = 2 \wedge s_2 = 1) \wedge s_3 = 1) \vee ((s_1 = 1 \wedge s_2 = 2) \wedge s_3 = 1)\\ \vee((s_1 = 2 \wedge s_2 = 2) \wedge s_3 = 2) \vee ((s_1 = 3 \vee s_2 = 3) \wedge s_3 = 0))).\end{aligned}$$

Any other sentence in the existential theory of the real numbers can be similarly translated into a sentence in the positive existential theory of the real numbers, with a linear increase in length. This sentence in the positive existential theory of the real numbers can then be translated into a sentence in PER-DNF as before, although with a possibly exponential increase in length.

Notice that to determine if sentences in PER-DNF are decidable, we just need to be able to determine if a set of polynomial equations are satisfiable over the positive reals. If there were a polynomial bound to the possible height of a solution (as an algebraic number) to such a set of polynomial equations, then the decidability question for PER-DNF would be in NP. Such an approach will not work, however, because it is possible to use a formula of PER-DNF to express a polynomial whose degree is exponential in the length of the formula.

We would now like to encode sentences from PER-DNF as diagrams. In order to do this, we will first consider a slightly more expressive form of diagram than those contained in FG. We will consider a system of diagrams that can contain a special length marker that marks dsegs of unit length. We will call such diagrams *unitized* diagrams. Unitized diagrams are more expressive than un-unitized diagrams, because in ordinary un-unitized diagrams, there is no way to force a segment to have unit length. Because ordinary diagrams only contain topological information and information about pieces being congruent, and because dilations and magnifications of a figure in the plane do not change its topology and preserve congruence, it follows that if a Euclidean plane M satisfies an ordinary diagram D, then any dilation or magnification of M about any point in the plane will also satisfy D. This is not true of unitized diagrams: if M satisfies a unitized primitive diagram D, then no dilation of M with scale factor $f \neq 1$ will satisfy D, because any segment that has length one in M will have length f in the dilation.

The reason that we need to consider unitized diagrams is that they will allow us to capture multiplication geometrically. We are going to encode real numbers as lengths of segments; then we will be able to represent addition of real numbers by putting segments next to one another. It is less obvious how to represent multiplication using lengths, though. We can use similar triangles to represent the fact that two ratios of lengths are equal; but we can only use this to represent multiplication if we have a unit length. Once we have a unit length, we can represent the fact that $a \times b = c$ by using similar triangles that show that $c/b = a/1$. And once we can represent both addition and multiplication, we should be able to represent all of PER-DNF.

We will show this in a way that is similar to what we did in Section 5. For every formula F of PER-DNF we will define a corresponding unitized diagram array $A(F)$ which is satisfiable if and only if F is. Assume that F contains numerals $n_1, \ldots n_j$, variable symbols $x_1, \ldots, x_q$, terms $t_1, \ldots, t_p$, and atomic formulae $f_1, \ldots, f_k$. Our diagram array will use $j + q + p + 1$ segment markers—one marker mn_i for each numeral n_i that occurs in F, one marker mx_i for each variable symbol x_i that occurs in F, one marker mt_i for each term t_i that occurs in F, and our designated unit marker. We will also need three additional angle markers am_{i1}, am_{i2}, and am_{i3} for each term t_i in F that is of the form $a \times b$.

We are now going to define one corresponding diagrammatic piece for each numeral, each term, and each atomic formula in F. We will do this as follows:

- If n_i is a numeral standing for the number N, then its corresponding diagrammatic piece is a dseg, divided into N pieces, each of which are marked with a unit marker, and the whole of which is marked by marker mn_i, as shown in Fig. 12 for the case $N = 4$.
- If t_i is a term, then there are four possibilities:
  (1) t_i might be a numeral n_r. In this case, its corresponding diagrammatic piece consists of a single dseg marked both by mt_i and by mn_r, as shown in Fig. 13.
  (2) t_i might be a variable symbol x_r. In this case, its corresponding diagrammatic piece consists of a single dseg marked both by mt_i and by mx_r, as shown in Fig. 14.
  (3) t_i might be of the form $a_i + b_i$. In this case a_i and b_i are both atomic terms and therefore each have their own corresponding markers ma_i and mb_i. The diagrammatic piece corresponding to t_i in this case is a dseg marked by mt_i, which is divided into two smaller dsegs marked by ma_i and mb_i, as shown in Fig. 15.
  (4) Finally, t_i might be of the form $a_i \times b_i$. In this case, its corresponding diagrammatic piece consists of two triangles. In both triangles, the three angles are marked by the three angle markers am_{i1}, am_{i2}, and am_{i3}. Two sides of the first triangle are marked by mt_i and mb_i, while the corresponding sides of the second triangle are marked by ma_i and the unit marker. This is shown in Fig. 16. (The idea behind this is that in any model of this diagrammatic piece, the two triangles must be similar because their angles are congruent, making their sides are proportional. This means that if the side marked by mt_i has length c, then $c/b = a/1$, so $c = ab$, as desired.)
- If f is an atomic formula of F then there are two possibilities:
  (1) f is of the form $t_i = t_r$. Then its corresponding diagrammatic piece consists of a single dseg, marked with the markers for both terms, mt_i and mt_r. This is shown in Fig. 17.
  (2) f is of the form $t_i < t_r$. Then its corresponding diagrammatic piece consists of a dseg marked with mt_r, divided into two smaller dsegs, one of which is marked by mt_i. This is shown in Fig. 18.

Next, recall that F is the existential closure of a disjunction of conjunctions; our array will contain one diagram $D(C)$ for each disjunctive clause C in F. Each such clause C is a conjunction of atomic formulae. $D(C)$ is defined to be the (smallest) diagram that contains as disjoint subdiagrams the diagrammatic pieces corresponding to each numeral, term, and atomic formula occurring in C. $A(F)$ is then the diagram array containing $D(C)$ for each disjunctive clause C in F. Note that $A(F)$ is again producible from F in log-space; in fact, we can produce it by just reading through F from left to right, producing the corresponding diagrammatic piece for each part of F as we come to it.

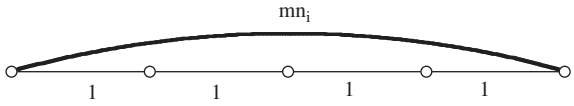

Fig. 12. The diagrammatic piece corresponding to the numeral n_i that stands for the number N.

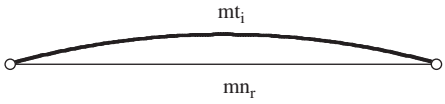

Fig. 13. The diagrammatic piece corresponding to t_i when t_i is the numeral n_i.

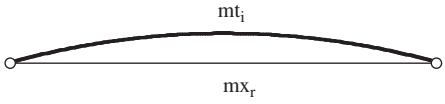

Fig. 14. The diagrammatic piece corresponding to t_i when t_i is the variable symbol x_i.

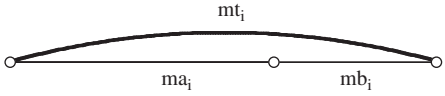

Fig. 15. The diagrammatic piece corresponding to t_i when t_i is the expression $a_i + b_i$.

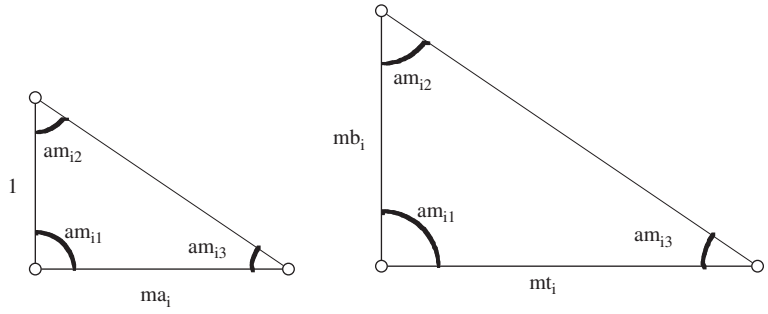

Fig. 16. The diagrammatic piece corresponding to t_i when t_i is the expression $a_i \times b_i$.

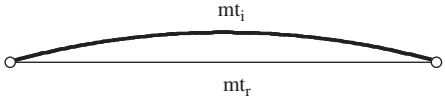

Fig. 17. The diagrammatic piece corresponding to f_i when f_i is the atomic formula $a_i = b_i$.

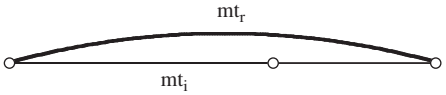

Fig. 18. The diagrammatic piece corresponding to f_i when f_i is the atomic formula $a_i < b_i$.

We are now in a position to prove the following theorem:

Theorem 6. *Let F be the formula $(\exists x_1 > 0) \cdots (\exists x_q)\, G(x_1, \ldots, x_q)$ of* PER-DNF, *where $G(x_1, \ldots, x_q)$ is quantifier free. Then the sequence of positive real numbers $(r_1, \ldots, r_q)$ satisfies $G(x_1, \ldots, x_q)$ if and only if there is a Euclidean Plane that satisfies F's corresponding unitized diagram array $A(F)$ in which the segments marked by mx_i have length r_i.*

Proof. ($\Rightarrow$) Assume that $(r_1, \ldots, r_q)$ satisfies $G(x_1, \ldots, x_q)$. Then, since G is a disjunction, there must be one of its disjunctive clauses C such that $(r_1, \ldots, r_q)$ satisfies C. Let $D(C)$ be C's corresponding diagram.

We can assign each term t in C a corresponding value $v(t)$, which is the value it inherits from the assignment of the values $(r_1, \ldots, r_q)$ to the variables $(x_1, \ldots, x_q)$. In particular, if a numeral n_i stands for the number N_i, then $v(n_i) = N_i$; $v(x_i) = r_i$; $v(a_i + b_i) = v(a_i) + v(b_i)$; and $v(a_i \times b_i) = v(a_i) \times v(b_i)$.

We will now construct a Euclidean Plane M that satisfies $D(C)$. M consists of the following disjoint pieces:

- For each numeral n_i occurring in C, M contains a segment of length $v(n_i)$, with points along the segment dividing it into $v(n_i)$ congruent pieces, each of length one. This part of M will satisfy the diagrammatic piece corresponding to n_i.
- For each term t_i occurring in C, M contains the following in order to satisfy the diagrammatic piece corresponding to t_i:
  (1) If t_i is a numeral n_j, then M contains a segment of length $v(n_j)$.
  (2) If t_i is a variable symbol x_j, then M contains a segment of length $v(x_j) = r_j$.
  (3) If t_i is of the form $a_i + b_i$, then M contains a new segment of length $v(t_i) = v(a_i) + v(b_i)$, divided into two smaller segments, one of length $v(a_i)$, and the other of length $v(b_i)$.
  (4) If t_i is of the form $a_i \times b_i$, then M contains two disjoint right triangles. In the first triangle, the side that is on the counterclockwise side of the right angle has length 1, and the side that is on the clockwise side of the right angle has length $v(a_i)$. In the second triangle, the side that is on the counterclockwise side of the right angle has length $v(b_i)$, and the side that is on the clockwise side of the right angle has length $v(t_i) = v(a_i) \times v(b_i)$. Note that each side of the second triangle is $v(b_i)$ times longer than the corresponding side of the first triangle. (This is true of the third sides by the Pythagorean Theorem.) Thus, all the sides of the triangles are in the same proportion, so the two triangles are similar and their angles are all the same, as $D(C)$ requires.
- For each atomic formula f_i occurring in C, M contains the following in order to satisfy the diagrammatic piece corresponding to f_i:
  (1) If f_i is of the form $a_i = b_i$, then M contains a segment of length $v(a_i)$. Note that in this case, we must have $v(a_i) = v(b_i)$, because the assignment of $(r_1, \ldots, r_q)$ to $(x_1, \ldots, x_q)$ makes C true, and f_i is one of the clauses of C, so it must be true under this assignment of values, which means that $v(a_i) = v(b_i)$.

(2) If f_i is of the form $a_i < b_i$, then M contains a segment of length $v(b_i)$, with a point along it marking off a segment of length $v(a_i)$. This is always possible, because as in the previous case, we must have $v(a_i) < v(b_i)$ because f_i must be true under the given assignment.

Finally, note that $M \models D(C)$ because it contains all the right pieces, and if two dsegs are labeled with the same label in $D(C)$, then they are actually the same length in M. In particular, if a dseg is marked by mn_i, then its corresponding segment in M has length $v(n_i)$; if a dseg is marked by mx_i, then its corresponding segment in M has length $v(x_i)$; if a dseg is marked by mt_i, then its corresponding segment in M has length $v(t_i)$; and if a dseg is marked by the unit marker, then its corresponding segment in M has length one. Thus, $M \models A(F)$ and gives each segment marked by mx_i the length $v(x_i) = r_i$, is as required.

($\Leftarrow$) Conversely, assume that $M \models A(F)$. Then M must satisfy one of the diagrams in this array, which must be $D(C)$ for some clause C of F. Since $M \models D(C)$, it must give all dsegs in $D(C)$ that are marked with a given marker the same length. So we can let r_1 be the length of those segments marked by mx_1, let r_2 be the length of those segments marked by mx_2, and so on. We need to show that the assignment of $(r_1, \ldots, r_q)$ to the variables $(x_1, \ldots, x_q)$ of C makes C true.

We can define $v(t)$ for each term t occurring in C exactly as in the previous case.

Our first step is to show that if a dseg in $D(C)$ is marked by marker mt_i, then its corresponding segment in M has length $v(t_i)$. This is true because:

- If t_i is a numeral, then the dseg marked with mt_i is also marked equal to a segment divided into $v(t_i)$ equal pieces, each one marked as having length one. (This segment comes from the numeral's corresponding diagrammatic piece.) So the segment in M that corresponds to this dseg must have length $v(t_i)$.
- If t_i is a variable, it is true by assumption that the segments marked by mt_i have length $r_i = v(t_i)$.
- If t_i is of the form $a_i + b_i$, then mt_i marks a segment S divided into two smaller segments marked with ma_i and mb_i. Since a_i and b_i must be numerals or variable symbols, it follows from the cases we have already shown and/or from our assumption that these two smaller segments must have lengths $v(a_i)$ and $v(b_i)$. Since the length of the whole segment S must be the sum of these two lengths, it must be equal to $v(a_i) + v(b_i) = v(a_i + b_i) = v(t_i)$.
- If t_i is of the form $a_i \times b_i$, then mt_i marks a dseg that is part of a pair of triangles like those shown in Fig. 16, which correspond to two triangles in M. Since all three pair of angles must be congruent in the two triangles, the triangles must be similar, and their sides must therefore be proportional. As before, we know from the previous cases and/or our assumption that those segments marked by ma_i and mb_i have lengths $v(a_i)$ and $v(b_i)$ respectively. So if c is the length of the side marked with mt_i, it follows from the similarity of the triangles that $c/v(a_i) = v(b_i)/1$. Cross multiplying, we have $c = v(a_i) \times v(b_i) = v(a_i \times b_i) = v(t_i)$, as required.

Now that we know that segments marked with mt_i have length $v(t_i)$, it follows almost immediately that each clause f_i of C must be true under the given assignment:

- If f_i is of the form $a_i = b_i$, then there is a dseg in $C(D)$ that is marked by both ma_i and mb_i. The corresponding segment in M must therefore have a length equal to both $v(a_i)$ and $v(b_i)$. The only way that this can happen is if $v(a_i) = v(b_i)$, which makes f_i true under the given assignment.
- On the other hand, if f_i is of the form $a_i < b_i$, then there is a dseg AB in $C(D)$ that is marked by mb_i and contains a smaller dseg AC that is marked by mb_i. AB's corresponding segment in M must therefore have a length equal to $v(b_i)$, and must contain a segment of length $v(a_i)$. As

before, the only way that this can happen is if $v(a_i) < v(b_i)$, which again makes f_i true under the given assignment.

Since each clause of C is true under the given assignment, C as a whole is true under the assignment, which makes G true under the assignment, as was required. □

This theorem shows that we can reduce questions about the satisfiability of formulas in PER-DNF to questions of satisfiability of unitized diagrams. We would really like to be able to use ordinary un-unitized diagrams, however, since diagrams in geometry do not normally contain a unit length marker. We can accomplish this using the following theorem:

Theorem 7. *Let D and D_u be primitive diagrams that are identical except that all instances of one dseg length marker m in the un-unitized diagram D are replaced by the unit length marker in the unitized diagram D_u. Then D is satisfiable if and only if D_u is satisfiable.*

Proof. ($\Rightarrow$) Assume that $M \vDash D$. Then all of the segments in M marked by marker m must be the same length l. Let M' be a Euclidean Plane obtained from M by dilating it about any point in the plane with scale factor $1/l$. Then M' has the same topology as M, and lengths and angles that were the same in M are the same in M', since dilations preserve shape and congruence. So M' still satisfies D. Furthermore, those segments in M' marked by marker m must have length $l \times (1/l) = 1$. Thus, they have unit length, so $M \vDash D_u$.

($\Leftarrow$) Assume that $M \vDash D_u$. D is identical to D_u, except that some segments that are required to have unit length by D_u are not required to have unit length by D. However, nothing in D says that they *cannot* have unit length; therefore $M \vDash D$. □

Taking the previous two theorems together, we arrive at the following corollary:

Corollary 8. *The satisfaction problem for* PER-DNF *is log-space many-one reducible to the diagram satisfaction problem.*

Conversely, in Section 6, we showed how to reduce the diagram satisfaction problem to that of the existential theory of the reals. Using the tricks discussed earlier in this section, we can convert the sentence of the existential theory of the reals that encodes a diagram into one in PER-DNF. As previously discussed, in the general case, a conversion like this can lead to an exponential growth in the length of the sentence. In this case, however, we have a finite bound to number of conjunctions that a disjunction can be inside, and this bound does not change as we increase the size of the diagram. Therefore, in this case, the sentence can be rewritten as one in PER-DNF with only a polynomial increase in length. This means that the diagram satisfaction problem is reducible to the satisfaction problem for PER-DNF in polynomial time. Combining this result with that of Corollary 8, we have the following conclusion:

Theorem 9. *The diagram satisfiability question and the satisfiability question for* PER-DNF *are in the same complexity class with respect to polynomial time many-one reductions.*

References

[1] T.L. Heath (Ed.), Euclid Elements, second ed., Dover Publications, New York, 1956.

[2] J. Hopcroft, J. Ullman, Introduction to Automata Theory, Languages, and Computation, Addison-Wesley, Reading, MA, 1979.
[3] N. Miller, A diagrammatic formal system for Euclidean geometry, Ph.D. Thesis, Cornell University, Ithaca, NY, 2001.
[4] N. Miller, Euclid and His Twentieth Century Rivals: Diagrams and the Logic of Euclidean Geometry (working title), CSLI Publications, Stanford, CA, to appear.
[5] J. Renegar, On the computational complexity and geometry of the first order theory of the reals, J. Symbolic Comput. 13 (1992).
[7] A. Tarski, A Decision Method for Elementary Algebra and Geometry, University of California Press, Berkeley, CA, 1951.