On Triangulations of a Set of Points in the Plane

by

Errol Lynn Lloyd

Massachusetts Institute of Technology
Laboratory for Computer Science
Cambridge, Massachusetts 02139

Abstract

A set, V, of points in the plane is triangulated by a subset T, of the straight-line segments whose endpoints are in V, if T is a maximal subset such that the line segments in T intersect only at their endpoints. The weight of any triangulation is the sum of the Euclidean lengths of the line segments in the triangulation. We examine two problems involving triangulations. We discuss the problem of finding a minimum weight triangulation among all triangulations of a set of points and give counterexamples to two published solutions to this problem. Secondly, we show that the problem of determining the existence of a triangulation, in a given subset of the line segments whose endpoints are in V, is NP-Complete.

1. Introduction

A recent development in complexity theory has been work dealing with computational geometry. A large portion of this work has been done by M. Shamos [10,11,12], who has given efficient algorithms for a number of fundamental geometric problems. These problems are important because many of the problems encountered in real-world circumstances involve sets of points in the plane. In the past, these problems have usually been abstracted out of the geometric domain into the realm of graph theory. These abstractions have potentially altered the essential nature of the problems, with the geometric constraints being lost. For instance, consider the problem of finding the minimum weight spanning tree of a set of points in the plane and the corresponding graph-theoretic problem of finding a minimum weight spanning tree of an arbitrary graph. It has been shown that the geometric problem, for n points in the plane, can be solved in time O(n log n), whereas the best algorithm for the graph-theoretic version requires time $O(n^2)$ for a graph with n vertices (in the worst-case) [1,10]. This suggests that the algebraic and geometric versions of a problem may have substantially different complexities. In contrast, both the algebraic and geometric versions of the Traveling Salesperson Problem and the Steiner Tree Problem have been shown to be NP-Complete [6,7,9]. Work in computational geometry is aimed at uncovering the mystery about what geometry contributes to a problem, as well as provide insights on the general nature of computation.

Our major result is an NP-Completeness proof involving the existence of triangulations. To our knowledge, with the exception of the results on Steiner Trees and Traveling Salesperson Tours, this is the only NP-Completeness result involving a problem specifically set in the plane.

2. The Triangulation Problems

The concept of a set of points in the plane being triangulated may be formulated as follows: Let V be a set of n distinct points in the plane. The points in V are vertices. Let L be the set of all straight-line segments between vertices in V. The elements of L are edges. Two edges, e and f, properly intersect if they intersect at a point which is not an endpoint of each, and e ≠ f. A triangulation of V is a maximal subset T, of L, such that no two edges of T properly intersect. This implies that each face of the straight-line planar graph determined by V and T is a triangle.

The minimum weight triangulation problem is as follows: Given a set of vertices, V, and the set of edges, L, whose endpoints are in V, a weight can be assigned to each edge in L, the weight of an edge being equal to the Euclidean distance between its endpoints. The weight of a triangulation, T, is then defined to be the sum of the weights of all of the edges in T. We are interested in an efficient algorithm for finding a triangulation of minimum weight among all of the triangulations of V. This problem will be referred to as MWT.

Triangulations have an application in the approximation of function values for a function of two variables when the value of the function is known at some number of arbitrary points [4]. The minimum weight triangulation problem has been studied previously by Duppe and Gottschalk [5] and Shamos and Hoey [10]. We note that other criteria for the "goodness" of a triangulation might be better suited to certain applications and may be easier to find. Criteria concerning the size of the maximum or minimum angles in a triangulation and how they apply to the finite element method are discussed by Babuska and Aziz [2] and Bramble and Zlamal [3].

The triangulation existence problem is: Given a set of vertices, V, and a subset E, of L, does there exist a subset T, of E, such that T is a triangulation of V? This problem will be referred to as TRI. An efficient algorithm for solving this problem might be useful in attacking other problems involving triangulations. For instance, in our work on MWT, we considered a matroid approach to the problem. A desirable property was to be able to tell efficiently if a subset of L contained a triangulation of V. It appears reasonable that other applications of triangulations may also have cause to use such an algorithm.

In section 3, we present counterexamples to two published algorithms for solving MWT. In section 4, we show that TRI is NP-Complete. The question of whether MWT is NP-Complete remains open.

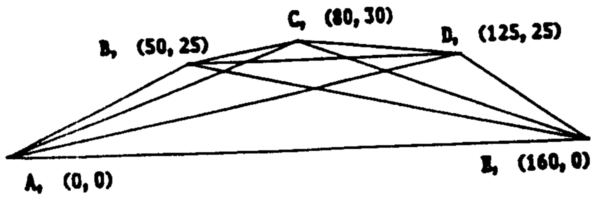
3. The MWT Problem

This section presents counterexamples to two algorithms conjectured to solve MWT.

The first of the algorithms is a classical "greedy" algorithm. This algorithm is as follows:

1. Let L_0 be the set of all edges with endpoints in V.
2. Set $T \leftarrow \emptyset$ and $i \leftarrow 0$.
3. While $L_i \neq \emptyset$ do
   31. Let w be an edge of least weight in L_i
   32. $T \leftarrow T \cup (w)$
   33. $L_{i+1} \leftarrow L_i - (w) - (m \in L_i \mid m$ and w properly intersect.)
   34. $i \leftarrow i+1$

Figure 1: Counterexample to the "greedy" algorithm.



Edge lengths of interior edges:

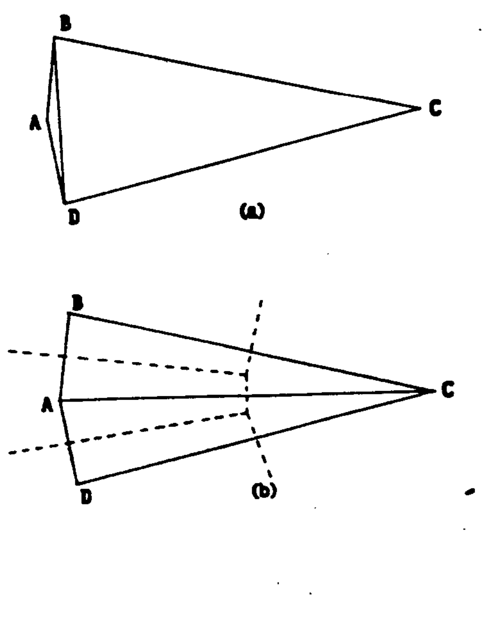
Edge BD: 75 units
Edge CE: <86 units
Edge AC: <86 units
Edge BE: >112 units
Edge AD: >127 units

The claim is that T is a minimum weight triangulation of V. In Figure 1 we give a set of vertices which shows that the triangulation produced is not necessarily a minimum weight triangulation. The edges not on the convex hull in the triangulation produced by this algorithm are BD and BE which have combined weight over 187 units. However, the interior edges in the minimum weight triangulation are CE and AC which have a combined weight of under 172 units. A variant of this algorithm was published as a solution to MWT by Duppe and Gottschalk [5]. The algorithm was independently suggested by R. L. Rivest.

The second algorithm purported to find minimum weight triangulations was given by Shamos and Hoey [10]. The essence of this algorithm is that the edges in the straight-line dual of the Voronoi diagram of V form a minimum weight triangulation of V. Figure 2 exhibits a set of vertices which show that this is not always the case. The minimum weight triangulation is shown in 2a and the triangulation produced by the Shamos-Hoey algorithm is given in 2b. The Voronoi edges are given as broken lines in 2b.

Several additional observations concerning minimum weight triangulations are given in [8].

Figure 2: Counterexample to the Shamos-Hoey algorithm



4. TRI is NP-Complete

In this section we show that TRI is NP-Complete. The major portion of the section is devoted to showing that the problem of conjunctive normal form satisfiability (CNF-SAT) is polynomially reducible to TRI. CNF-SAT is NP-Complete [7].

4.1 Intuition and Overview

Assume that we have clauses $C_1, C_2, \ldots, C_k$ each of which is a sum of literals drawn from the variables $x_1, x_2, \ldots, x_n$. The problem is to determine if there is a truth assignment to the variables such that each clause is satisfied. From the k clauses we will construct a set of vertices, V, and a set of edges, E, whose endpoints are in V, such that there is a subset T, of E, triangulating V if and only if the set of clauses is satisfiable. Throughout this section a triangulation of V will refer to a subset T, of E, whose edges are a triangulation of V.

The building block in our construction will be a set of vertices and edges which we will refer to as a switch. A rectangular array of these switches will be employed, with one switch for each clause-variable pair. This array of switches will also be called the network. We let S_{ij} represent the switch for variable x_i and clause C_j. Each switch S_{ij} will be one of three types depending on whether x_i is in C_j or $\bar{x}_i$ is in C_j or neither is in C_j. We note that the switches are numbered in an x-y fashion as opposed to standard matrix numbering.

In any triangulation of this array of switches, we may regard two streams of information to be flowing through each switch, one stream flowing vertically and the other from left to right horizontally. The vertical stream of information flowing through S_{ij} carries a truth value for variable x_i. For each x_i the same truth value must be flowing vertically through each switch S_{ij}, $1 \le j \le k$. The horizontal stream of information leaving switch S_{ij} on the right indicates whether or not clause C_j is satisfied by the assignment of the truth values (as determined by vertically flowing information for each variable) to the variables $x_1, \ldots, x_i$. This information may then flow into the left side of switch $S_{i+1,j}$. Our construction forces the information flowing into the left side of each switch S_{1j} to be "not satisfied" and the information flowing out of the right side of each switch S_{nj} to be "satisfied". Thus, the horizontal information for clause C_j can flow completely across the network only if the truth assignment to some x_i satisfies C_j, causing the horizontal flowing information about C_j to change from "not satisfied" to "satisfied" in switch S_{ij}.

4.2 Description of a Switch

Before giving a formal specification of the sets V and E we will describe the structure of a switch. Each switch will consist of the vertices and edges given in Figure 3. Note that the coordinates are given relative to E_1 as the origin. In Figure 4 is a pictorial representation of a switch. An enlarged view of the center portion of a switch is shown in Figure 5.

Various vertices of each switch are classified as follows:

Frame vertices: E_1, E_2, E_3, E_4, F, G, H, I, J, L, M, N, P, Q, R, S
Terminals: $A_1, A_2, B_1, B_2, C_1, C_2, D_1, D_2$
Matched pair of terminals: A_1 and A_2, B_1 and B_2, C_1 and C_2, D_1 and D_2

Figure 3: Switch Specifications

Each switch consists of the following vertices. The coordinates of each vertex are given relative to E_1.

E_4 (0, 100)	L (37, 100)					J (63, 100)	E_3 (100, 100)
M (0, 63)	S (37, 63)					R (63, 63)	I (100, 63)
			A_2 (47, 57)	D_1 (53, 57)			
		B_2 (43, 53)			C_1 (57, 53)		
		C_2 (43, 47)			B_1 (57, 47)		
			D_2 (47, 43)	A_1 (53, 43)			
N (0, 37)	P (37, 37)					Q (63, 37)	H (100, 37)
E_1 (0, 0)	F (37, 0)					G (63, 0)	E_2 (100, 0)

Each switch consists of the following edges:

Frame Edges: E_1F, E_1N, FP, FN, NP, E_2G, E_2H, GH, GQ, HQ,
E_3I, E_3J, IJ, IR, JR, E_4L, E_4M, LM, LS, MS

Non-frame Edges: FR, GS, HM, HS, IN, IP, JP, LQ, MQ, NR,
A_1G, A_1Q, A_1H, A_1I, A_1C_1, A_1A_2, A_1S, A_1B_2, A_1C_2, A_1D_2, A_1P, A_1F,
B_1G, B_1Q, B_1H, B_1I, B_1R, B_1L, B_1D_1, B_1A_2, B_1M, B_1C_2, B_1N, B_1D_2, B_1P, B_1F,
C_1Q, C_1H, C_1I, C_1R, C_1J, C_1L, C_1S, C_1A_2, C_1M, C_1B_2, C_1N, C_1D_2, C_1F,
D_1H, D_1I, D_1R, D_1J, D_1L, D_1S, D_1A_2, D_1B_2, D_1C_2, D_1P, D_1D_2,
A_2Q, A_2R, A_2J, A_2L, A_2S, A_2M, A_2N, A_2C_2,
B_2H, B_2I, B_2R, B_2J, B_2L, B_2S, B_2M, B_2N, B_2P, B_2G, B_2D_2,
C_2Q, C_2H, C_2I, C_2J, C_2S, C_2M, C_2N, C_2P, C_2F, C_2G,
D_2G, D_2Q, D_2R, D_2M, D_2N, D_2P, D_2F

When it is appropriate we will superscript the vertices of a switch. For example, N^{ij} is vertex N in switch S_{ij}. Note that each switch is symmetric in structure with respect to the lines $x = 50$ and $y = 50$ (the lines relative to E_1). The notations AB and [A, B] will each be used to refer to an edge whose endpoints are vertices A and B. If Q and R are sets of vertices in V, then Q×R represents the set of all edges [q, r] such that q is in Q and r is in R.

4.3 Specification of the Sets V and E

4.3.1 The Basics

As previously stated, our construction consists of a rectangular array of switches with one switch S_{ij} for each variable x_i, clause C_j pair. Adjacent switches in this network will coincide on appropriate frame vertices. Vertex E_1 of switch S_{ij} will have coordinates $(100\cdot(i-1), 100\cdot(j-1))$.

To insure that the convex hull of V is in E we modify the switches in the outermost rows and columns of the network. These switches will be identical to regular switches except they will have one additional vertex (called a special vertex) and several additional edges. These special switches are specified as follows:

1. Each switch S_{1j}, contains a special vertex, T^{1j}, with coordinates $(0, 100\cdot(j-1)+50)$ and the edges $(T^{1j})\times(M^{1j}, N^{1j}, A_2^{1j}, B_2^{1j})$

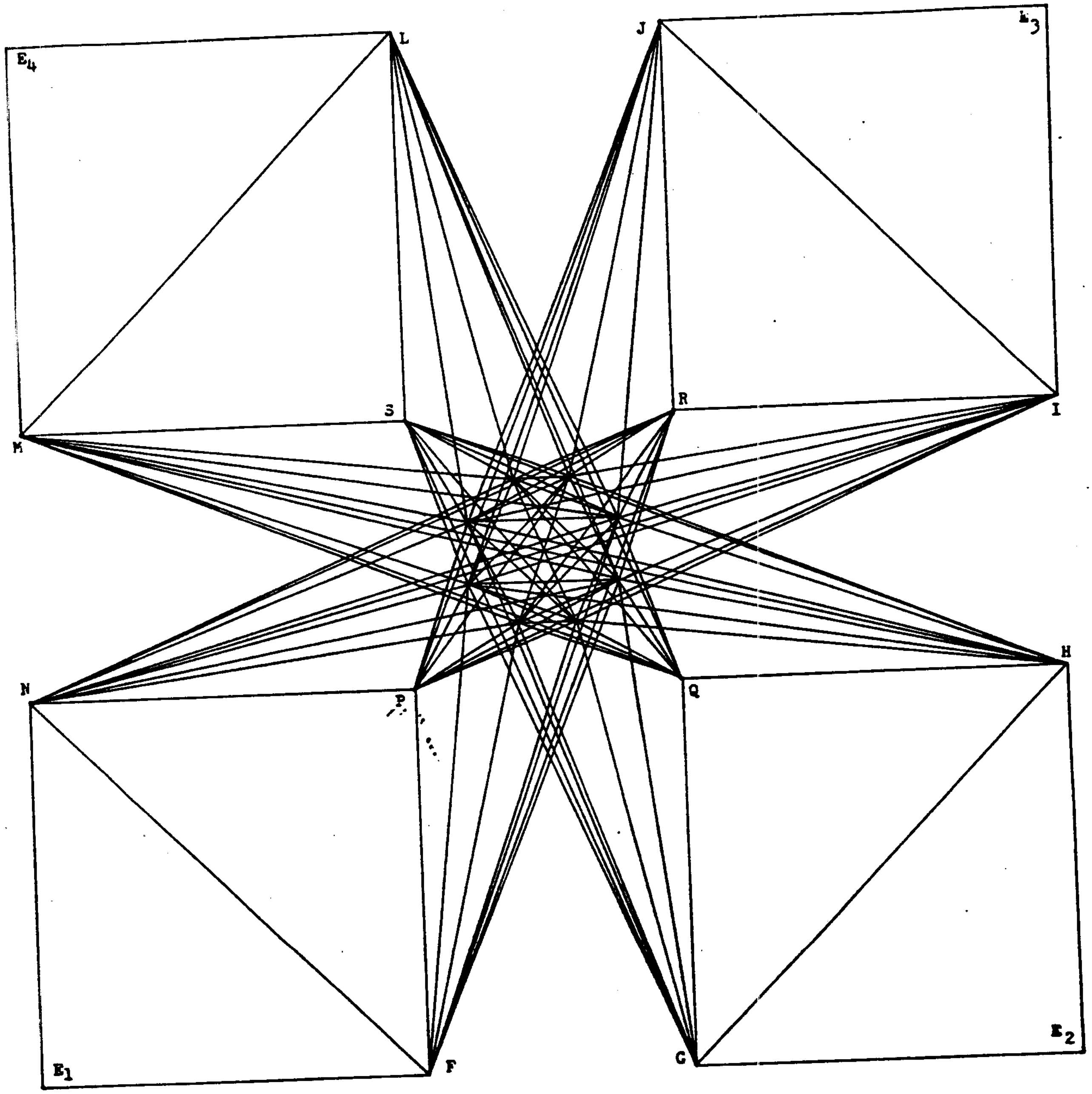
2. Each switch S_{i1}, contains a special vertex, U^{i1}, with coordinates $(100\cdot(i-1)+50, 0)$ and the edges $(U^{i1})\times(F^{i1}, G^{i1}, A_1^{i1}, B_1^{i1}, C_2^{i1}, D_2^{i1})$

3. Each switch S_{nj}, contains a special vertex, V^{nj}, with coordinates $(100n, 100\cdot(j-1)+50)$ and the edges $(V^{nj})\times(H^{nj}, I^{nj}, C_1^{nj}, D_1^{nj})$

4. Each switch S_{ik}, contains a special vertex, W^{ik}, with coordinates $(100\cdot(i-1)+50, 100k)$ and the edges $(W^{ik})\times(J^{ik}, L^{ik}, A_2^{ik}, B_2^{ik}, C_1^{ik}, D_1^{ik})$

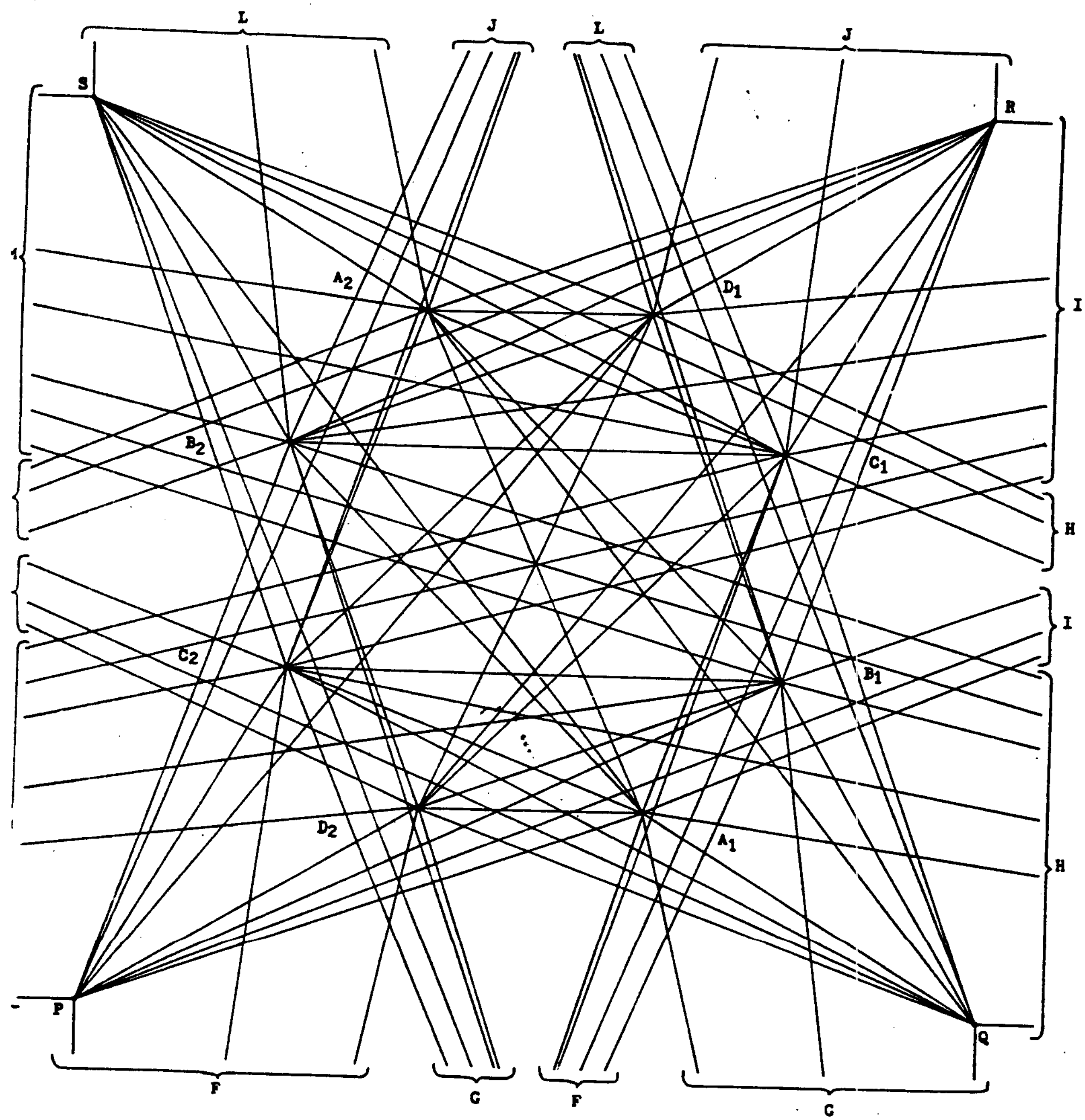
The frame is defined to be a set consisting of the frame edges of each switch in the network, as well as each edge of the network which has a frame vertex as one endpoint and a special vertex as the other endpoint. We note that no edge with a terminal as an endpoint is included in the frame.

Figure 4: A Switch



The eight unlabeled vertices in the center portion of the switch are the terminals. Figures 3 and 5 show the labels of these vertices.

Figure 5: An enlarged view of the center portion of a switch



4.3.2 The Interswitch Edges

In addition to the edges within each switch there are edges in E whose endpoints lie in different switches. These edges will be called interswitch edges. Only terminals will be endpoints of interswitch edges and these edges will lie only between adjacent switches. It will be shown later that between any (horizontally or vertically) adjacent pair of switches, exactly one interswitch edge will be present in any triangulation of V. Intuitively, the chosen edge will carry information from one switch to the other.

Vertical interswitch edges are specified as follows: For each i and j pair, with $1 \le i \le n$ and $1 \le j < k$, the following edges are placed in E:

$$(A_2^{ij}, C_1^{ij}) \times (A_1^{i,j+1}, C_2^{i,j+1}) \quad \text{and}$$
$$(B_2^{ij}, D_1^{ij}) \times (B_1^{i,j+1}, D_2^{i,j+1})$$

Intuitively, these edges will carry the vertical flowing information about the truth values of the variables, with the A-C edges carrying "false" and the B-D edges carrying "true".

The horizontal interswitch edges between two adjacent switches S_{ij} and $S_{i+1,j}$ will vary depending on the nature of switch S_{ij}. For this reason we classify each switch as being one of three possible types:

S_{ij} is a neutral switch if $x_i \notin C_j$ and $\bar{x}_i \notin C_j$

S_{ij} is a positive switch if $x_i \in C_j$

S_{ij} is a negative switch if $\bar{x}_i \in C_j$

Horizontal interswitch edges may be specified as follows:

1. For each i and j pair, with $1 \le i < n$ and $1 \le j \le k$, such that S_{ij} is a neutral switch the following edges are placed in E:

$$(A_1^{ij}, B_1^{ij}) \times (A_2^{i+1,j}, B_2^{i+1,j}) \quad \text{and}$$
$$(C_1^{ij}, D_1^{ij}) \times (C_2^{i+1,j}, D_2^{i+1,j})$$

We define terminals A_1 and B_1 to be Clause-false and terminals C_1 and D_1 to be Clause-true in a neutral switch. Intuitively, these interswitch edges and the edges specified below, will carry the horizontal flowing information about the clauses, with edges with a Clause-false endpoint carrying "not satisfied" and edges with a Clause-true endpoint carrying "satisfied."

2. For each i and j pair, with $1 \le i < n$ and $1 \le j \le k$, such that S_{ij} is a positive switch the following edges are placed in E:

$$(A_1^{ij}) \times (A_2^{i+1,j}, B_2^{i+1,j}) \quad \text{and}$$
$$(B_1^{ij}, C_1^{ij}, D_1^{ij}) \times (C_2^{i+1,j}, D_2^{i+1,j})$$

We define terminal A_1 to be Clause-false and terminals B_1, C_1 and D_1 to be Clause-true in a positive switch.

3. For each i and j pair, with $1 \le i < n$ and $1 \le j \le k$, such that S_{ij} is a negative switch the following edges are placed in E:

$$(B_1^{ij}) \times (A_2^{i+1,j}, B_2^{i+1,j}) \quad \text{and}$$
$$(A_1^{ij}, C_1^{ij}, D_1^{ij}) \times (C_2^{i+1,j}, D_2^{i+1,j})$$

We define terminal B_1 to be Clause-false and terminals A_1, C_1 and D_1 to be Clause-true in a negative switch.

Additionally, based on the type (positive, negative or neutral) of each switch S_{nj}, the following edges are included in S_{nj}:

1. For each j, with $1 \le j \le k$, such that S_{nj} is a positive switch, edge $[V^{nj}, B_1^{nj}]$ is placed in E.

2. For each j, with $1 \le j \le k$, such that S_{nj} is a negative switch, edge $[V^{nj}, A_1^{nj}]$ is placed in E.

4.3.3 The Sets V and E

Set V contains all frame vertices, terminals and special vertices of each switch in the network.

Set E contains all of the edges of each switch in the network, as well as the interswitch edges as specified in the previous section. We note that the frame is included in E and that no edge in E properly intersects any edge of the frame. This means that any triangulation of V must contain all of the edges in the frame.

Finally, we note that the construction can be done in time polynomial in n and k. There are n k switches in the network. Each switch may be constructed in a constant amount of time. Interswitch edges exist only between adjacent pairs of switches. There are O(n·k) such pairs. The vertical interswitch edges are the same for each adjacent pair of switches, hence, they can be constructed in constant time for any given pair. The horizontal interswitch edges for any pair of adjacent switches depend only on the type of the left switch in the pair and, hence, can be constructed in constant time for any given pair of switches. Thus, the sets V and E can be constructed in time O(n·k).

4.4 Proof that a solution to TRI yields a solution to CNF-Satisfiability

In this section we assume that T is a subset of E and is a triangulation of V. We show that there is a truth assignment to the variables $x_1, \ldots, x_n$ such that each clause $C_1, \ldots, C_k$ is satisfied. This truth assignment will be obtained from T.

4.4.1 Preliminaries

As stated earlier the frame must be included in T. This means that the non-frame edges in T must:

1. Complete the triangulation of each switch in the network.
2. Connect the switches together in a manner which triangulates V.

As we shall show, T must fulfill these conditions with a very particular structure.

A terminal, α, in switch S_{ij} is defined to be East-connected in triangulation T if and only if there exists an edge $\alpha\beta$ in T such that $\alpha\beta$ properly intersects edge $[I^{ij}, H^{ij}]$. Now consider edge $[I^{ij}, H^{ij}]$. Since this edge is not in E, hence not in T, there must be an edge in T which properly intersects $[I^{ij}, H^{ij}]$. By our construction, each such edge has a terminal of S_{ij} as an endpoint. This means that there must be at least one East-connected terminal per switch in any triangulation of V. Similarly, we can define and imply the existence of at least one West-connected, one North-connected, and one South-connected terminal per switch in any triangulation of V. A connected terminal is a terminal that is at least one of East-connected, West-connected, North-connected or South-connected.

In the proof of the switch triangulation theorem we will make use of the following property of triangulations:

If edge y_1y_2 is in triangulation T and is not on the convex hull of V, then y_1y_2 is an edge in the boundary between two of the triangular faces of the straight-line planar graph determined by V and T (one face in each half-plane as determined by the line through y_1 and y_2).

This property will be used in the following proof as follows: In general, there will be an edge y_1y_2 in T and a specified half-plane. Consider the set of vertices, P, such that for each vertex w in P:

1. w lies in the specified half-plane.
2. Edges y_1w and y_2w are in E.
3. No other vertex of V lies on, or interior to, triangle y_1y_2w.

If there is only one vertex w, in P, then edge y_1y_2 in T forces edges y_1w and y_2w to be in T by the property of triangulations stated above. This is denoted by $y_1y_2 \rightarrow y_1y_2w$.

If there are two vertices, z_1 and z_2, in P then we will use the following notation:

$y_1y_2 \rightarrow$ choice

1. $y_1y_2z_1$
   ...
2. $y_1y_2z_2$
   ...

Typically, the first choice of $y_1y_2z_1$ will lead to a situation where an edge r is forced to be in T and yet there is already an edge s in T such that r and s properly intersect. Such a contradiction will be denoted "# to s". In the proof in the next section an edge is said to be <u>finally enumerated</u> if it doesn't lead to a contradiction if placed in T. It may be that $|P| \geq 2$ and no vertex in P leads to a contradiction, but, that there exists a vertex y_3 in V such that for each w in P, $y_1w \rightarrow y_1wy_3$. Intuitively, edge y_1y_2 in T forces edge y_1y_3 into T but the "force" requires two steps. In this case we write $y_1y_2 \overset{2}{\rightarrow} y_1y_3$.

4.4.2 The Switch Triangulation Theorem

<u>Theorem 1</u>: Given any triangulation of V there are exactly two connected terminals in each switch and, furthermore, for each switch those two terminals are a matched pair of terminals.

<u>Proof</u>

Consider any triangulation T, of V, and any switch S_{ij} in the network. At least one terminal of S_{ij} is East-connected. Only terminals A_1, B_1, C_1 and D_1 in S_{ij} may be East-connected.

<u>Case 1</u>: Suppose terminal A_1 is East-connected in S_{ij}. Then there is a vertex Z in V such that A_1Z is in T and A_1Z properly intersects line segment IH of S_{ij}. By our construction Z is one of $A_2^{i+1,j}$, $B_2^{i+1,j}$, $C_2^{i+1,j}$ or $D_2^{i+1,j}$ if i < n or is V^{nj} if i = n. Then, in S_{ij},

$A_1Z \rightarrow A_1ZH$

$A_1H \rightarrow A_1HQ$

$A_1Z \rightarrow A_1ZI$

$A_1I \rightarrow A_1IP$

$A_1P \rightarrow A_1PF$

A_1Q and A_1F force A_1G

$IP \rightarrow IPB_1$

$PB_1 \rightarrow PB_1D_2$

$IB_1 \rightarrow$ choice

1. IB_1R

   $B_1R \rightarrow B_1RF$ # to PB_1

2. IB_1C_2

   $B_1C_2 \rightarrow$ choice

   1. B_1C_2H # to PB_1
   2. B_1C_2N

      $B_1N \rightarrow B_1ND_2$

      $ND_2 \rightarrow ND_2P$

      $IC_2 \rightarrow IC_2N$

      $IN \rightarrow INC_1$

      $NC_1 \rightarrow NC_1B_2$

      $C_1B_2 \rightarrow$ choice

      1. C_1B_2M

         $C_1M \rightarrow C_1MA_2$

         $C_1A_2 \rightarrow C_1A_2S$

         $C_1S \rightarrow C_1SH$ # to A_1I

      2. C_1B_2I

         $B_2I \rightarrow B_2ID_1$

         $B_2D_1 \rightarrow B_2D_1R$

         $B_2R \rightarrow B_2RN$

         $RN \rightarrow RNA_2$

         $RA_2 \rightarrow RA_2J$

         $A_2N \overset{2}{\rightarrow} A_2M$

         $A_2M \rightarrow A_2MS$

         A_2S and A_2J force A_2L

∴ If A_1 is East-connected then A_1 is South-connected and A_2 is North-connected and West-connected. Furthermore, A_1 and A_2 are the only connected terminals in S_{ij}.

Because of the symmetries of the switch we also have:

1. If D_1 is East-connected then D_1 is North-connected and D_2 is South-connected and West-connected.

2. If A_2 is West-connected then A_2 is North-connected and A_1 is South-connected and East-connected.

3. If D_2 is West-connected then D_2 is South-connected and D_1 is North-connected and East-connected.

Furthermore, in each of the above three cases, the terminals in the matched pair of connected terminals are the only connected terminals in S_{ij}.

The non-interswitch edges which are finally enumerated in the above proof (along with the frame edges of S_{ij}) constitute a triangulation of S_{ij}. This triangulation is called an <u>A-triangulation</u> and is pictured in Figure 6. In an A-triangulation we say that terminal A_1 is East-exposed and South-exposed and terminal A_2 is West-exposed and North-exposed.

Analogously, corresponding to D_1 and D_2 being the connected terminals of S_{ij}, there is a set of non-interswitch edges called a <u>D-triangulation</u>. This triangulation is shown in Figure 7. In a D-triangulation terminal D_1 is East-exposed and North-exposed and terminal D_2 is West-exposed and South-exposed.

<u>Case 2</u>: Suppose terminal B_1 is East-connected in S_{ij}. Then there is a vertex Z in V such that B_1Z is in T and B_1Z properly intersects line segment IH of S_{ij}. Because of our construction Z is one of $A_2^{i+1,j}$, $B_2^{i+1,j}$, $C_2^{i+1,j}$ or $D_2^{i+1,j}$ if $i<n$ or is V^{nj} if $i=n$. Then, in switch S_{ij}, $B_1Z \rightarrow B_1ZI$. Now consider which terminal is West-connected in S_{ij}. From case 1, since B_1 is East-connected it cannot be A_2 or D_2. Suppose it is C_2. Then C_2M and C_2N must be in T. Then,

$C_2M \rightarrow$ choice

- 1. C_2MB_1
  - $MB_1 \rightarrow MB_1H$ # to B_1I
- 2. C_2MS
  - $C_2S \rightarrow C_2SG$
  - $B_1I \rightarrow$ choice
    - 1. B_1IC_2
      - $IC_2 \rightarrow IC_2N$ # to C_2M
    - 2. B_1IR
      - $B_1R \rightarrow B_1RF$ # to SG

$\therefore$ C_2 is not West-connected, hence, B_2 is West-connected.

Now, in switch S_{ij},

$B_1Z \rightarrow B_1ZI$

$B_1Z \rightarrow B_1ZH$

$B_1I \rightarrow$ choice

- 1. B_1IC_2
  - $IC_2 \rightarrow IC_2N$
  - $C_2N \rightarrow$ choice
    - 1. C_2NP
      - $C_2P \rightarrow$ choice
        - 1. C_2PD_1 # to IC_2
        - 2. C_2PF
          - $C_2F \overset{2}{\rightarrow} C_2G$
          - $C_2G \rightarrow C_2GS$ # to IC_2
    - 2. C_2NB_1
      - $NB_1 \rightarrow NB_1D_2$
      - $B_1D_2 \rightarrow B_1D_2P$
      - $B_1P \rightarrow B_1PI$ # to B_1H
- 2. B_1IR
  - $B_1R \rightarrow B_1RF$
  - $B_1F \overset{2}{\rightarrow} B_1G$
  - $B_1G \rightarrow B_1GQ$

$\therefore$ B_1 is the only East-connected and the only South-connected terminal.

Furthermore, since B_2 is West-connected, by the symmetries of the switch, analogously to the above, we can show that B_2 is the only West-connected and the only North-connected terminal. This shows that non-frame edges B_2M, B_2S, B_2L, B_2J, B_2N, B_2P and PJ are also in T. All that remains is to show that the region bordered by the vertices P, J, R and F can indeed be triangulated. This can be done with edges JR, JC_2, C_2P, C_2A_2, A_2J, A_2D_1, D_1J, D_1R, D_1D_2, D_2R, D_2C_1, C_1R, C_1F, C_1A_1, A_1F, A_1D_2, D_2F, PF, D_2P, D_1P and C_2D_1.

$\therefore$ If B_1 is East-connected then B_1 is South-connected and B_2 is North-connected and West-connected. Furthermore, B_1 and B_2 are the only connected terminals in S_{ij}.

The non-interswitch edges which are finally enumerated in the proof of case 2 (along with the frame edges of S_{ij}) constitute a <u>B-triangulation</u>. This triangulation is shown in Figure 8. In a B-triangulation, terminal B_1 is East-exposed and South-exposed and terminal B_2 is West-exposed and North-exposed.

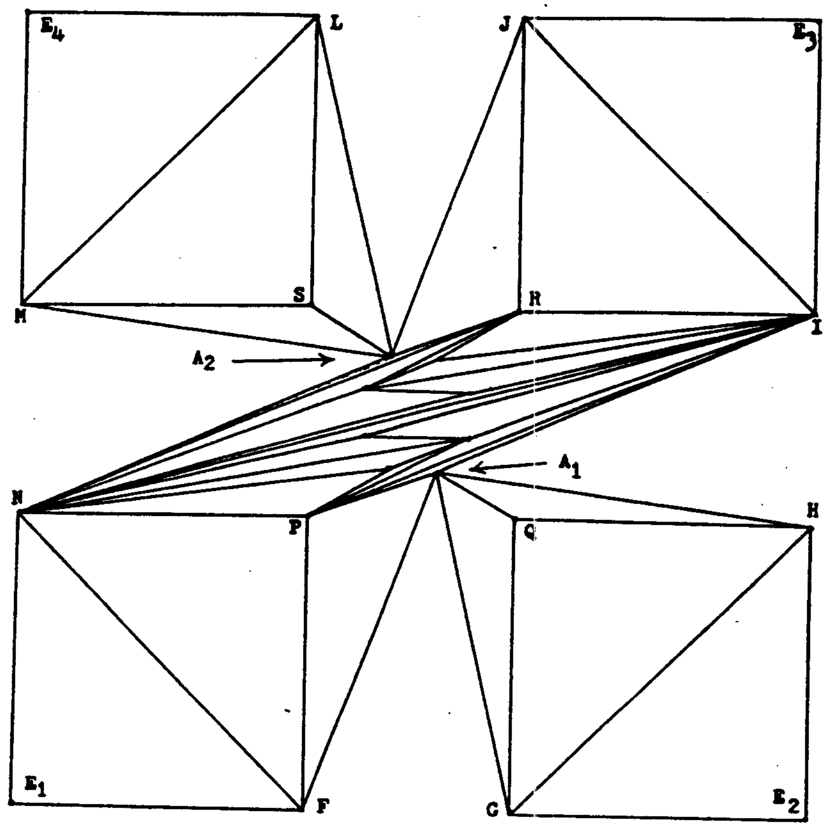
Because of the symmetry of the switch we also have: If C_1 is East-connected then C_1 is North-connected and C_2 is West-connected and South-connected. Furthermore, C_1 and C_2 the only connected terminals in S_{ij}. Correspondingly, there is a set of non-interswitch edges called a <u>C-triangulation</u>. This triangulation is shown in Figure 9. In a C-triangulation, terminal C_1 is East-exposed and North-exposed and terminal C_2 is West-exposed and South-exposed.

$\therefore$ Given any triangulation of V there are exactly two connected terminals per switch and they are a matched pair of terminals. □

The following corollary follows immediately from the above theorem and our earlier remarks about the non-frame edges in T:

<u>Corollary 1</u>: If S_1 and S_2 are adjacent switches in the network and T is a triangulation of V, then there is exactly one interswitch edge in T whose endpoints are a terminal in S_1 and a terminal in S_2.

Figure 6: The A-triangulation



The edges in an A-triangulation are:

Each frame edge,

A_1P, A_1F, A_1G, A_1Q, A_1H, A_1I,

B_1I, B_1C_2, B_1N, B_1D_2, B_1P,

C_1I, C_1B_2, C_1N,

D_1I, D_1R, D_1B_2,

A_2R, A_2J, A_2L, A_2S, A_2M, A_2N,

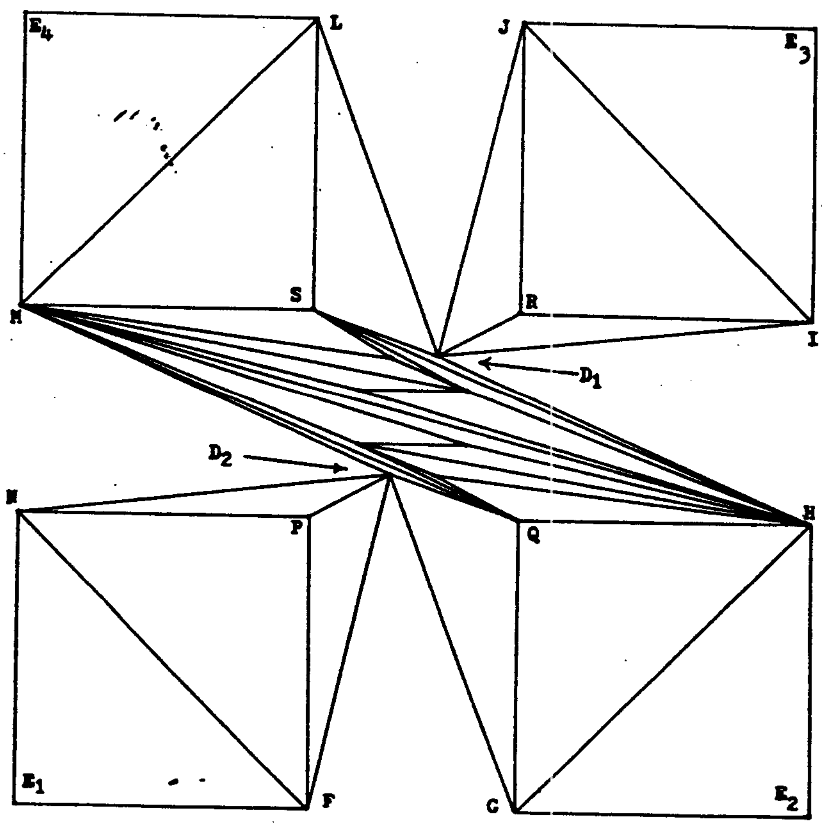
B_2R, B_2N, B_2I,

C_2I, C_2N,

D_2N, D_2P,

IP, NR, IN.

Figure 7: The D-triangulation



The edges in a D-triangulation are:

Each frame edge,

A_1Q, A_1H, A_1C_2,

B_1H, B_1M, B_1C_2,

C_1H, C_1S, C_1M, C_1A_2, C_1B_2,

D_1H, D_1I, D_1R, D_1J, D_1L, D_1S,

A_2S, A_2M,

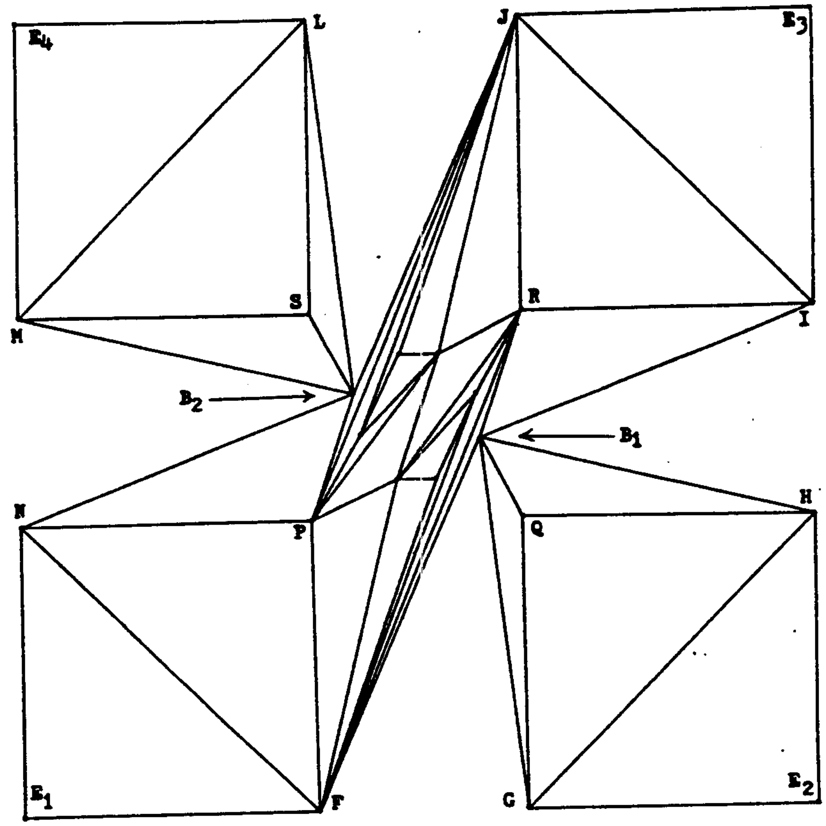
B_2H, B_2M,

C_2M, C_2Q, C_2H,

D_2M, D_2N, D_2P, D_2F, D_2G, D_2Q,

MS, MQ, HM.

Figure 8: The B-triangulation



The edges in a B-triangulation are:

Each frame edge,

A_1F, A_1C_1, A_1D_2,

B_1F, B_1G, B_1Q, B_1H, B_1I, B_1R,

C_1F, C_1R, C_1D_2,

D_1R, D_1J, D_1A_2, D_1C_2, D_1P, D_1D_2,

A_2J, A_2C_2,

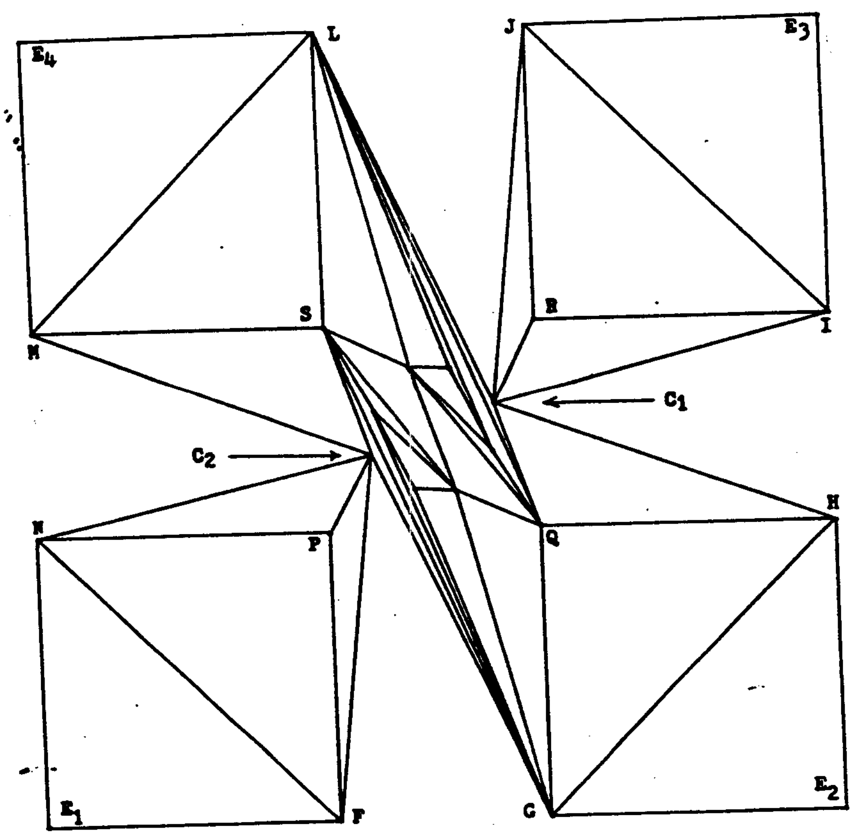
B_2J, B_2L, B_2S, B_2M, B_2N, B_2P,

C_2J, C_2P,

D_2R, D_2P, D_2F,

FR, JP.

Figure 9: The C-triangulation



The edges in a C-triangulation are:

Each frame edge,

A_1G, A_1Q, A_1A_2, A_1S, A_1B_2, A_1D_2,

B_1Q, B_1L, B_1D_1, B_1A_2,

C_1Q, C_1H, C_1I, C_1R, C_1J, C_1L,

D_1L, D_1A_2,

A_2L, A_2S, A_2Q,

B_2S, B_2G, B_2D_2,

C_2S, C_2M, C_2N, C_2P, C_2F, C_2G,

D_2G,

GS, LQ.

4.4.3 The Main Result

In the specifications of interswitch edges we defined various terminals to be Clause-true and Clause-false. For convenience, those definitions are restated here:

> In a neutral switch, terminals A_1 and B_1 are Clause-false and terminals C_1 and D_1 are Clause-true.
>
> In a positive switch, terminal A_1 is Clause-false and terminals B_1, C_1 and D_1 are Clause-true.
>
> In a negative switch, terminal B_1 is Clause-false and terminals A_1, C_1 and D_1 are Clause-true.

The following three lemmas are useful in proving the main result and follow directly from our constructions, Theorem 1 and Corollary 1:

Lemma 1: In any given triangulation of V, for each i, $1 \leq i \leq n$, either the connected terminals are B's and D's for all S_{ij}, or the connected terminals are A's and C's for all S_{ij}, $1 \leq j \leq k$.

Lemma 2: In any given triangulation of V, the West-connected terminal in each switch S_{1j} is A_2^{1j} or B_2^{1j} and the East-connected terminal in each switch S_{nj} is Clause-true, for $1 \leq j \leq k$.

Lemma 3: In any given triangulation of V, for each j, $1 \leq j \leq k$, there exists an i, with $1 \leq i \leq n$, such that the East-connected terminal of S_{ij} is either A_1 or B_1 and it is Clause-true.

Now consider the following truth assignments to the variables $x_1,\ldots,x_n$:

> x_i is true if the South-connected terminal in S_{i1} is B_1 or D_2.
>
> x_i is false if the South-connected terminal in S_{i1} is A_1 or C_2.

Theorem 2: For each j, $1 \leq j \leq k$, the clause C_j is satisfied by this truth assignment to the variables.

Proof

Consider any j such that $1 \leq j \leq k$. By Lemma 3, there is an i such that the East-connected terminal of S_{ij} is either A_1 or B_1 and it is Clause-true.

Case 1: The connected terminal is B_1. Since it is Clause-true this must be a positive switch, so x_i is in C_j. But then B_1 is the South-connected terminal and by Lemma 1, the South-connected terminal of S_{i1} is B_1 or D_2. Then by our assignment x_i is true and C_j is satisfied.

Case 2: The connected terminal is A_1. Since it is Clause-true this must be a negative switch, so $\bar{x}_i$ is in C_j. But then A_1 is the South-connected terminal and by Lemma 1, the South-connected terminal of S_{i1} is A_1 or C_2. Then, by our assignment x_i is false and C_j is satisfied. □

Therefore, from a triangulation T, of V, with T a subset of E, we have obtained a truth assignment to the variables $x_1, \ldots, x_n$ such that each of the clauses $C_1, \ldots, C_k$ is satisfied.

4.5 Proof that a solution to CNF-Satisfiability yields a solution to TRI

Assume that $H_1,\ldots,H_n$ is a truth assignment to $x_1,\ldots,x_n$ such that each of the clauses $C_1,\ldots,C_k$ is satisfied. We will show that there is a subset T, of E, such that the edges in T triangulate V. Initially we note that a set, T, consisting of edges meeting the following requirements will suffice as a triangulation of V. It is clear that T need only include:

1. The edges in the frame.

2. The edges in a triangulation of each switch in the network. That is, for each switch, the edges in either an A, B, C or D-triangulation.

3. For each adjacent pair of switches an edge whose endpoints are the appropriate exposed terminals of those switches. (The exposed terminals having been determined by the triangulations specified in 2.)

4. For each special vertex in V, an edge whose endpoints are the special vertex and the appropriate exposed terminal of the switch in which the special vertex is located.

The remainder of this section is devoted to specifying a set of edges which meets the above requirements. Initially we place the frame in T and again note that no edge in E properly intersects any edge in the frame. The frame edges thus present no further difficulty.

4.5.1 The Triangulation of Each Switch

For each clause, C_j, we define W_j to be the least i such that x_i is in C_j or $\bar{x}_i$ is in C_j and the truth assignment of H_i to x_i causes C_j to be satisfied. Then, switch S_{ij} is triangulated in T as follows:

For $i \leq W_j$, if H_i is true
then S_{ij} is B-triangulated
else S_{ij} is A-triangulated.

For $i > W_j$, if H_i is true [redacted]
then S_{ij} is D-triangulated
else S_{ij} is C-triangulated.

The exposed terminals of a switch are determined by the triangulation specified for that switch.

4.5.2 Interswitch Edges in T

Theorem 3: For each i and j pair, with $1 \leq i \leq n$ and $1 \leq j \leq k-1$, there is an edge in E whose endpoints are the North-exposed terminal of S_{ij} and the South-exposed terminal of $S_{i,j+1}$.

Proof

The result follows directly from our construction of vertical interswitch edges and the specifications given above for the triangulations of each switch. □

Theorem 4: For each i and j pair, with $1 \leq i \leq n-1$ and $1 \leq j \leq k$, there is an edge in E whose endpoints are the East-exposed terminal of S_{ij} and the West-exposed terminal of $S_{i+1,j}$.

Proof

Consider any i and j pair such that $1 \leq i \leq n-1$ and $1 \leq j \leq k$.

Case 1: $i > W_j$

Because $i > W_j$, the East-exposed terminal of S_{ij} is either C_1^{ij} or D_1^{ij} and the West-exposed terminal of $S_{i+1,j}$ is either $C_2^{i+1,j}$ or $D_2^{i+1,j}$. But, by our interswitch edge specifications each of the four edges: $[C_1^{ij}, C_2^{i+1,j}]$, $[C_1^{ij}, D_2^{i+1,j}]$, $[D_1^{ij}, C_2^{i+1,j}]$, and $[D_1^{ij}, D_2^{i+1,j}]$ is in E.

Case 2: $i = W_j$

Subcase 1: The East-exposed terminal of S_{ij} is B_1^{ij}. By the definition of W_j this switch is either a positive or negative switch. Assume that it is a negative switch, hence $\bar{x}_i$ is in C_j. But since B_1^{ij} is the East-exposed terminal, H_i is true. This contradicts the definition of W_j. Therefore, this is a positive switch. Since $i+1 > W_j$ the West-exposed terminal of $S_{i+1,j}$ is either $C_2^{i+1,j}$ or $D_2^{i+1,j}$. But, by our interswitch edge specifications both of the edges $[B_1^{ij}, C_2^{i+1,j}]$ and $[B_1^{ij}, D_2^{i+1,j}]$ are in E.

Subcase 2: The East-exposed terminal of S_{ij} is A_1^{ij}. Similarly to subcase 1 we can show that this is a negative switch and that the desired edge exists in E.

9 **Case 3:** $i < W_j$

Subcase 1: The East-exposed terminal of S_{ij} is B_1^{ij}.

Subcase a: Switch S_{ij} is a neutral switch. Because $i+1 \leq W_j$, the West-exposed terminal of $S_{i+1,j}$ is either $A_2^{i+1,j}$ or $B_2^{i+1,j}$. By the interswitch specifications both of the edges $[B_1^{ij}, A_2^{i+1,j}]$ and $[B_1^{ij}, B_2^{i+1,j}]$ are in E.

Subcase b: Switch S_{ij} is a positive switch. This means that x_i is in C_j. Because B_1^{ij} is the East-exposed terminal of S_{ij}, the truth value of H_i is true. But this means that C_j is satisfied by the assignment of H_i to x_i. This is a contradiction of the definition of W_j. Hence, S_{ij} is not a positive switch.

Subcase c: Switch S_{ij} is a negative switch. Because $i+1 \leq W_j$, the West-exposed terminal of $S_{i+1,j}$ is either $A_2^{i+1,j}$ or $B_2^{i+1,j}$. But, by our interswitch edge specifications involving S_{ij}, a negative switch, each of the edges: $[B_1^{ij}, A_2^{i+1,j}]$ and $[B_1^{ij}, B_2^{i+1,j}]$ is in E.

Subcase 2: The East-exposed terminal of S_{ij} is A_1^{ij}. The proof is analogous to that of subcase 1, with the roles of subcases b and c reversed. □

Hence, for each pair of adjacent switches there is an edge in E whose endpoints are the appropriate exposed terminals of those switches. Each of these edges is placed into T.

4.5.3 Additional Special Switch Edges in T

Theorem 5: For each special vertex in V there is an edge in E whose endpoints are the special vertex and the appropriate exposed terminal of the switch in which the special vertex is located.

Proof

Case 1: The special vertex is V^{nj} with $1 \leq j \leq k$.

Subcase 1: $n > W_j$

Because $n > W_j$, the East-exposed terminal of S_{nj} is either C_1^{nj} or D_1^{nj}. By our basic specifications of special switches each of the edges $[V^{nj}, C_1^{nj}]$ and $[V^{nj}, D_1^{nj}]$ is in E. Thus, whichever terminal is East-exposed in S_{nj} the desired edge is in E.

Subcase 2: $n = W_j$

Subcase a: The East-exposed terminal of switch S_{nj} is B_1^{nj}. Then, from case 2 of the proof of Theorem 4, this is a positive switch. But by our additional edge specifications in section 4.3.2, the edge $[V^{nj}, B_1^{nj}]$ is in E.

Subcase b: The East-exposed terminal of switch S_{nj} is A_1^{nj}. Then, from case 2 of the proof of Theorem 4, this is a negative switch. But by our additional edge specifications in section 4.3.2, the edge $[V^{nj}, A_1^{nj}]$ is in E.

Case 2: The special vertex is U^{i1}, W^{ik} or T^{1j}. It follows directly from our basic specifications of special switches that the desired edge is in E. □

Hence, for each special vertex in V there is an edge in E whose endpoints are the special vertex and the appropriate exposed terminal of the switch that the special vertex is a part of. Each of these edges is placed in T.

We have now specified a set of edges T, which is a subset of E and which satisfies the four requirements given as being sufficient for a triangulation of V. Hence, the set T is a triangulation of V.

This completes the proof that CNF-Satisfiability is polynomially reducible to TRI.

4.6 Finishing Up

Theorem 6: TRI is NP-Complete.

Proof

In the first 5 subsections of this section we have shown that CNF-Satisfiability, a known NP-Complete problem, is polynomially reducible to TRI. All that remains to show is that TRI is in NP. Consider an instance of TRI as specified by the sets V and E. We know that a set T is a triangulation of V if and only if the following two properties hold for T:

1. No two edges in T properly intersect.

2. For every edge, e, whose endpoints are vertices of V, either e is in T or e properly intersects some edge in T.

Hence, given the sets V and E, we non-deterministically choose the set T and then verify that these two properties hold. To test for property 1 requires time $O(|T|^2)$ and testing for property 2 may be done in time $O(|V|^2|T|)$. Therefore, TRI is in NP and hence, TRI is NP-Complete. □

5. Conclusion

We have shown that TRI is NP-Complete. The major open question is to resolve the status of MWT. Consider the two Hamiltonian circuit problems and the two spanning tree problems corresponding to these two triangulation problems. Both of the corresponding Hamiltonian circuit problems (that is, the problem of existence given some of the edges and the problem of minimum weight given all of the edges) are NP-Complete. In comparison, there are efficient algorithms for both of the spanning tree problems. Given that TRI is NP-Complete, we conjecture that MWT is also NP-Complete (actually, NP-Hard as we have stated it).

References

1. Aho, A.V., Hopcroft, J.E. and Ullman, J.D., The Design and Analysis of Computer Algorithms. Addison-Wesley (1974), 470 pp.

2. Babuska, I. and Aziz, A.K., "On the Angle Condition in the Finite Element Method," SIAM Journal on Numerical Analysis. 13:2 (1976) 214-226.

3. Bramble, J.H. and Zlamal, M., "Triangular Elements in the Finite Element Method," Mathematics of Computation. 24 (1970), 809-820.

4. Davis, J.C. and McCullagh, M.J., eds. Display and Analysis of Spatial Data. Wiley (1975), 378 pp.

5. Duppe, R.D. and Gottschalk, H.J., "Automatische Interpolation von Isolinien bei willkurlich stutzpunkten," Allgemeine Vermessungsnachrichten. 77 (1970), 423-426.

6. Garey, M.R., Graham, R.L. and Johnson, D.S., "Some NP-Complete Geometric Problems," Proceedings of the Eighth ACM Symposium on the Theory of Computing. (1976), 10-22.

7. Karp, R.M., "Reducibility Among Combinatorial Problems," in Complexity of Computer Computations, R.E. Miller and J.W. Thatcher, eds., Plenum Press (1972), 85-104.

8. Lloyd, E.L., "On Triangulations of a Set of Points in the Plane," M.I.T. Laboratory for Computer Science Technical Memo No. MIT/LCS/TM 88 (1977).

9. Papadimitriou, C.H., "The Euclidean Traveling Salesman Problem is NP-Complete," Princeton University Computer Science TR No. 191 (1975).

10. Shamos, M.I. and Hoey, D. "Closest Point Problems," Proceedings of the Sixteenth Annual Symposium on the Foundations of Computer Science. IEEE (1975), 151-162.

11. Shamos, M.I., "Geometric Complexity," Proceedings of the Seventh ACM Symposium on the Theory of Computing. (1975), 224-233.

12. Shamos, M.I., Problems in Computational Geometry. (1974) unpublished manuscript.

This research was supported by National Science Foundation Grants MCS76-14294 and MCS74-12997-A04