THE EQUIVALENCE OF THEOREM PROVING AND THE INTERCONNECTION PROBLEM

James F. Lynch
Department of Mathematics
University of Colorado
Boulder, Colorado 80302
and
Storage Technology Corporation
Louisville, Colorado 80027

ABSTRACT

A simple method of converting statements in mathematical logic to equivalent interconnection problems is demonstrated. This shows that if there exists a practical method for solving interconnection problems, then there exists a practical proof procedure for statements in mathematical logic. (In the terminology of Cook [1] and Karp [4], the interconnection problem is NP-complete.) Since such a procedure has been sought for without success for almost 100 years, we should not expect the interconnection problem to be any easier. This holds equally true for finding a method which will route a given percentage of wires.

I. INTRODUCTION

We shall phrase the interconnection problem in graph theoretic terminology. $G = \langle N,E \rangle$ will represent an undirected graph with node set N and edge set E. Individual edges will be denoted by two element sets giving the nodes they connect.

The graph theoretic model of the interconnection problem is as follows. Given a graph $G = \langle N,E \rangle$ and disjoint subsets $S_1,\ldots,S_k$ of N, find disjoint connected subsets $T_1,\ldots,T_k$ of N such that $S_i \subseteq T_i$ for $i = 1,\ldots,k$. Here G represents the circuit board with N the set of possible endpoints of wires on the board and E the set of allowable pathways between points. The sets $S_1,\ldots,S_k$ are the endpoints of nets 1 through k respectively, and the sets $T_1,\ldots,T_k$ are the points in a routing of nets 1 through k.

This formalization of the interconnection problem will be related to statements in the propositional calculus, i.e., boolean formulas of variables. A basic problem with such formulas is determining whether a given formula is satisfiable, i.e., whether there exists an assignment of True or False (1 or 0) to the variables such that the formula is True (takes on the value 1) for that assignment.

The problem of finding a practical procedure for determining satisfiability of a formula in propositional calculus (or the dual problem of *proving* a formula is a tautology) has been studied as far back as Frege [2] in the 1870's, and it appears to be unsolvable. We shall show how, given a formula in propositional calculus, it can be recoded into an interconnection problem such that the formula is satisfiable if and only if the interconnection problem is solvable. Thus, if there were a practical method for solving the interconnection problem, there would be a practical method for solving the satisfiability problem. This should cast serious doubt on the existence of a good interconnection algorithm, even, as we shall see, one which merely routes a given percentage of the wires.

II. ENCODING FORMULAS INTO GRAPHS

We begin by showing how to encode a formula into an interconnection problem on a graph.

Theorem I. Given a formula σ of the propositional calculus, there is a graph with an interconnection problem which is solvable if and only if σ is satisfiable.

Proof. By a result of Cook [1] σ can easily be converted to conjunctive normal form (CNF); thus we may assume $\sigma = C_1 \wedge \ldots \wedge C_m$, and for $i=1,\ldots,m$ $C_i = w_{i1} \vee \ldots \vee w_{in_i}$ where each w_{ij} is a literal, i.e., a *variable or the negation of a variable. The remainder of the proof shows how to convert* σ, *given* that it is in CNF, into an interconnection problem.

We construct $G = \langle N,E \rangle$ as follows. Let σ be as above, with $v_1,\ldots,v_n$ the distinct variables occurring in σ. (Refer to Figure 1.) N will consist partly of the following nodes: nodes C_i and C_i' for each clause C_i, and nodes v_j and v_j' for each variable v_j. There will be exactly two paths between each v_j and v_j', called the high and low paths. The remaining nodes will be on these paths.

Now consider each clause C_i and the variables occurring in it. If v_j is a literal in C_i (occurring without the negation sign), place the node n_{ij1} on the low path from v_j to v_j', add it to N, and add the edges $\{C_i, n_{ij1}\}$ and $\{C_i', n_{ij1}\}$ to E. If $\overline{v}_j$ (negation of v_j) is in C_i, place n_{ij0} on the high path from v_j to v_j', add it to N, and add $\{C_i, n_{ij0}\}$, $\{C_i', n_{ij0}\}$ to E. Finally, add edges to E to explicitly form all the high and low paths.

We must now specify the subsets S_i of N which are to be connected.

$S_i = \{C_i, C_i'\}$ for $i = 1,\ldots,m$ and

$S_{m+j} = \{v_j, v_j'\}$ for $j = 1,\ldots,n$.

Assuming σ is satisfiable, we must show the existence of $T_1,\ldots,T_{m+n}$. Let $f: \{v_1,\ldots,v_n\} \to \{0,1\}$ be the satisfying assignment of the variables. For $j = 1,\ldots n$ let T_{m+j} be the high path between v_j and v_j' if $f(v_j) = 1$, and let it be the low path if $f(v_j) = 0$.

To construct T_i for $i = 1,\ldots,m$ note that since σ is satisfied by f, C_i contains some literal which takes on the value 1, i.e., at least one of the following cases holds:

(1) for some j, v_j occurs as a literal in C_i and $f(v_j) = 1$

(2) for some j, $\overline{v}_j$ occurs as a literal in C_i and $f(v_j) = 0$

If case (1) holds, let $T_i = \{C_i, n_{ij1}, C_i'\}$. Then $T_i \cap T_{m+j} = \emptyset$ since T_{m+j} is the high path and n_{ij1} is on the low path. By similar reasoning, if case (2) holds, let $T_i = \{C_i, n_{ij0}, C'_i\}$.

By the construction of G, all the T_i's are connected and pairwise disjoint, and $S_i \subseteq T_i$ for $i = 1, \ldots, m+n$. Thus the existence of a satisfying assignment for σ implies the existence of $T_1,\ldots,T_{m+n}$ with the the desired properties.

Conversely, given $T_1,\ldots,T_{m+n}$ define

$$f(v_j) = 1 \text{ if } T_{m+j} \text{ is the high path}$$
$$= 0 \text{ if } T_{m+j} \text{ is the low path}$$

for $j = 1,\ldots,n$.

Then it is easy to see that (1) or (2) above holds for each clause C_i. Therefore σ is satisfiable.

This completes the proof.

It is interesting to note that the graph G of the theorem can be embedded in a two layer circuit board where each layer is a rectangular grid. Thus, it is conceivable that there are circuit boards in existence that "prove" fairly nontrivial theorems in the propositional calculus.

III. FURTHER RESULTS

It is even possible that there are one layer circuit boards that are encodings of difficult problems in mathematical logic. This is contained in the following theorem.

Theorem II. Given a formula σ of the propositional calculus, there is an interconnection problem on a planar graph which is solvable if and only if σ is satisfiable.

Proof. To avoid tedious details we proceed informally, using the example of Figure 1. Basically we will show how to remove nonplanar parts of the graph while retaining the property that the graph is routable if and only if the formula σ is satisfiable.

One type of nonplanarity is illustrated in Figure 1, where a path from C_1 to C_1' crosses over the high and low paths from v_1 to v_1'.

Figure 2 shows how the nonplanarity is removed. The set $S_1 = \{v_1, v_1'\}$ is also replaced by the sets $\{v_1, x_1\}$, $\{x_2, x_3\}$, and $\{x_4, v_1'\}$. These additional nodes and edges permit a connection to be made from C_1 to C_1' while insuring that the same type of path (high or low) is taken at v_1 and v_1'.

The other type of nonplanarity is also illustrated in Figure 1, where a path from C_1 to C_1' intersects the high path from v_2 to v_2' but crosses over the low path. This may be removed in a manner very similar to the method described above.

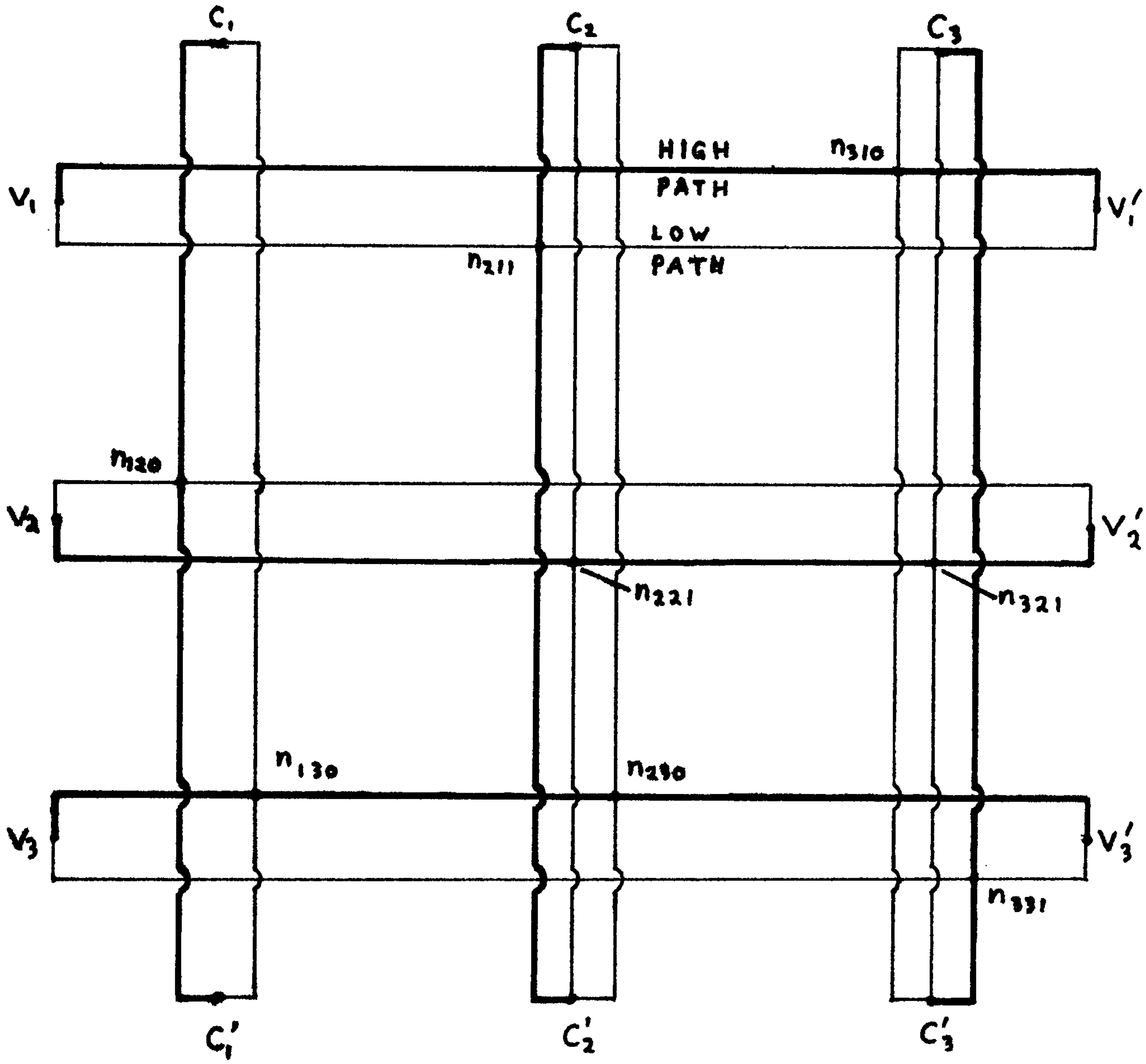
When all nonplanarities have been removed, we have a planar graph G' which is routable if and only if G is satisfiable, thus completing the informal proof.

Again, note that G' is embeddable in a rectangular grid.

We have shown that completely routing a circuit board is a very difficult problem, but one may still believe that routing only a certain number or percentage of the connections might be much easier. This is not the case, however. Garey, Johnson, and Stockmeyer [3] have shown that the problem of satisfying a given number of clauses in a formula in CNF is essentially the same problem as satisfying all the clauses, i.e., satisfying the formula. A slight modification of our proof carries the result over to the interconnection problem, showing that routing a given percentage of the wires is still very difficult.

REFERENCES

1. Cook, S.A. "The Complexity of Theorem Proving Procedures." Proceedings of the Third Annual ACM Symposium on Theory of Computing, 1971, pp. 151-158.
2. Frege, G. Begriffsschrift, eine der Arithmetischen Nachgebildete Formelsprache des Reinen Denkens, Halle, 1879.
3. Garey, M.R., Johnson, D.S. and Stockmeyer, L. "Some Simplified NP-Complete Problems." Proceedings of the Sixth Annual ACM Symposium on Theory of Computing, 1974, pp.47-63.
4. Karp, R.M. "Reducibility Among Combinatorial Problems." Complexity of Computer Computations, Miller and Thatcher, eds., Plenum Press, 1972, pp.85-104.



$\sigma = C_1\ C_2\ C_3,$

$C_1 = \bar{v}_2 V \bar{v}_3$

$C_2 = v_1 V v_2 V \bar{v}_3$

$C_3 = \bar{v}_1 V v_2 V v_3$

satisfying assignment

$f(v_1) = 1$

$f(v_2) = 0$

$f(v_3) = 1$

Figure 1.

Example of a formula σ and its encoding as a graph. The heavy lines connect the sets $T_1, \ldots, T_6$ resulting from the satisfying assignment.

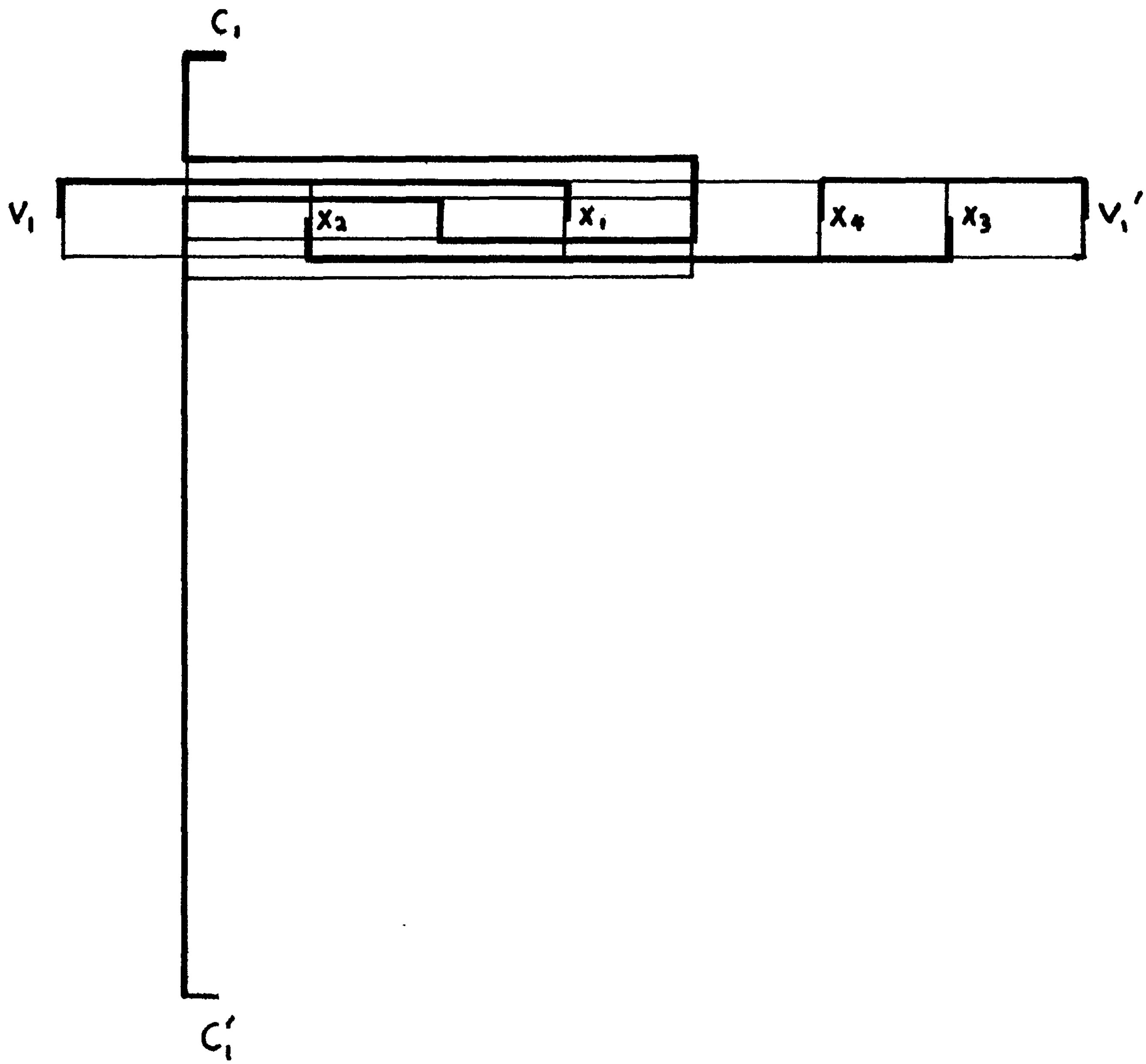


Figure 2.

Removal of nonplanarity from Figure 1. The heavy lines show the high path taken at v_1 and v_1' and the path between C_1 and C_1'.