Fast Loop Matrix Generation for Hybrid Analysis and a Comparison of the Sparsity of the Loop Impedance and MNA Impedance Submatrices

Vrinda S Ovalekar  
H Narayanan*

Abstract

Since the time computers began to be used for analysis of circuits, loop analysis methods have been regarded as inefficient. This is because, for most practical circuits, the number of loop equations is comparable to the number of node equations but the loop impedance matrix has usually much larger density than the node admittance matrix. However, almost two decades ago, extremely fast and powerful algorithms became available for mesh matrix generation of planar graphs. Use of such methods allows hybrid methods (partly node, partly loop analysis) to compete with modified nodal analysis. To take an extreme situation, if the circuit permits impedance but not conductance formulation, it is possible for most practical topologies (nearly planar circuits) to produce loop coefficient matrices which have a much smaller size and a lower number of entries than the corresponding modified nodal coefficient matrix. This paper reports on a new method developed for loop matrix generation which is essentially an extension of the mesh matrix generation method, to handle nonplanar graphs. This method, based on Tarjan and Hopcroft's planarity testing algorithm, was found to yield sparse loop impedance matrices; for graphs (size  > 500 nodes, 1500 edges) with a nonplanarity of 10%, the density of the resulting loop impedance matrix was less than 1%. For graphs with nonplanarities less than 2%, the number of non-zero entries in the loop impedance matrix obtained with this method were found to be less than 5×number of edges of the graph. This method of loop matrix generation has been incorporated into the general purpose hybrid circuit simulator BITSIM being developed at the VLSI Design Centre, IIT Bombay.

1 Introduction

The rapid development in the area of circuit design has resulted in an increasing dependence on circuit simulation for accurate analysis of circuits. Analysis time of general purpose circuit simulators like SPICE is very large, when used for large scale circuits. Various methods have been tried to improve the speed of circuit simulation, for such circuits, over the last few years.

A possible direction in which improvement in speed can be sought is through use of iterative methods for solving the linear equations which result at each Newton Raphson iteration at each time point. However iterative methods are very sensitive to the structure of the coefficient matrices. The commonly used Modified Nodal Analysis (MNA) yields matrices whose principal minors may be singular even for passive RLMC circuits.

At the VLSI Design Centre, IIT, Bombay, a general purpose circuit simulator BITSIM [1] based on a hybrid equation formulation method [2] is being developed. Here, elements of the original network are partitioned into admittance type and impedance type of elements, to form an admittance and an impedance network. Nodal analysis and loop analysis, respectively, are then used on the admittance and impedance networks. In this case, the resulting matrix for the original network appears to have a good structure from the point of view of iterative methods (in particular, for passive RLMC circuits, the coefficient matrix turns out to be positive definite).

Various methods of loop analysis were tried out for the impedance network in BITSIM. Standard methods such as the fundamental circuit matrix method yield very dense coefficient matrices which is unsatisfactory both from the point of view of storage as well as speed. We have developed a new method to obtain a loop matrix such that the resulting loop impedance matrix is always sparse. This method is an extension of the method of mesh matrix generation used for planar networks, to a general network. The graph representing a nonplanar network is repeatedly sliced into planar subgraphs using Tarjan and Hopcroft's planarity algorithm [3] and meshes for each of these are constructed, to finally give a loop matrix. As this method uses the property of planarity, the number of non-zero entries in the loop impedance matrix in this case increases with increase in nonplanarity of the network. For most networks with nonplanarities of about 2%, the number of non-zero entries was found to be less than that for the MNA matrix. In these cases loop analysis is expected to perform better than MNA for iterative methods of linear equation solution, both the number of iterations and the time taken for each iteration of the linear equation solution being less than that for MNA.

This paper describes the new method and discusses the results obtained using this and other more familiar methods on random test graphs.

The organisation of the paper is as follows. Section

*Deptt. of Elec. Engg., IIT, Bombay 400076, INDIA

0-7803-0593-0/92 $3.00 1992 IEEE
2 gives a description of the new loop matrix generation method. Section 3 contains the experimental results obtained.

2 The New Loop Matrix Generation Method

This section describes the new method of loop matrix generation which is essentially an extension of the mesh matrix generation method for planar networks. Here the graph representing a nonplanar network is repeatedly sliced into planar subgraphs and meshes for each of these are obtained to finally give a loop matrix.

The extraction of planar slices and the generation of meshes for these slices was done using Tarjan and Hopcroft's planarity testing algorithm [3] and the mesh generation algorithm described by Melhorn [4].

The nonplanar edges are handled as follows. The Tarjan and Hopcroft's algorithm stops on encountering the first nonplanar edge. We made additions to the algorithm to obtain a larger planar slice; on finding a nonplanar edge, the algorithm now continues after removing the edge and marking it as nonplanar. All such marked edges are added along with a tree to form a new subgraph $G_2$. The tree used is a minimum spanning tree built on the vertices of the last planar slice, using edges of least weight from all those previously embedded in the plane, except those edges which were also nonplanar edges of the last planar slice. The weight assigned to each edge is equal to the number of planar slices to which the edge belonged. Once the new subgraph $G_2$ is formed, the algorithm for planar slice extraction and mesh generation is applied to it. This iterative process is continued till the new subgraph formed is planar.

The validity of the slicing technique rests on the following theorem.

**Theorem:** Let $G$ be a graph. Let $G_1$, be a subgraph of $G$ on the set of edges $X$. Let $V_1$ be the subset of vertices of $G_1$ at which edges in $E(G) - X$ are incident. Let $t$ be a connected sub-tree of $G_1$ whose vertices contain $V_1$. Let $G_2$ be the subgraph of $G$ on $(E(G) - X) \cup t$. Let $L_1, L_2$ be independent and complete set of loops for writing the KVL Equations of $G_1, G_2$ respectively. Then $L_1 \cup L_2$ is an independent and complete set of loops for $G$.

This method of loop matrix generation gives sparse loop impedance matrices.

3 Experimental Results

The new loop matrix generation method was tested on a number of graphs. The programs for all the loop matrix generation methods were written in standard C and run on the SUN 3/260 system, in an UNIX environment. This section gives the results obtained with this method.

To generate the test graphs for the loop matrix methods, a random graph generator program was developed. This program first constructed random maximal planar graphs (i.e., planar graphs with triangular meshes and outermost region) [3]. Then depending on the nonplanarity required, additional edges were added at random to this graph.

We define, Nonplanarity % of a graph on $V$ vertices as $\frac{\text{Nonplanar edges} \times 100}{\text{Edges of a maximal planar graph on } V \text{ vertices}}$.

The computation times for the new method, for graphs of different sizes and a nonplanarity of 10% are given in Table 1. It can be seen that the computation time for loop matrix generation is small compared to the total analysis time; a graph on 5000 vertices and 16494 edges took 8 minutes, 19 seconds (on the SUN 3/260 system) to form a loop matrix while the expected time of analysis of a dynamic circuit on such a graph is several hours.

The density of the $BBT$ matrix (number of non-zero entries / total number of entries) was calculated using an additional routine. The results obtained with the new method as well as the standard fundamental circuit matrix method for graphs of different non-planarities are given in Table 2. As the new method uses the property of planarity, the number of non-zero entries in the loop impedance matrix in this case increase with increase in nonplanarity of the network. But even for nonplanarity of 10%, the sparsity (1-density) obtained with the new method is much more than that obtained with the fundamental circuit matrix method. For graphs (size > 500 nodes, 1500 edges) with a nonplanarity of 10%, the density of the loop impedance matrix in this case was less than 1%.

The loop matrix generation method is to be used in the hybrid simulator BITSIM for impedance network analysis. As such, to compare this method with MNA, the matrix generated by MNA for an impedance network should be considered. In this case, the equations will consist of KCL and branch equations. The KCL equations at each vertex except the reference vertex will contribute entries equal to the degree of the vertex, to the matrix; the total number of entries due to KCL equations will be equal to $(2 \times \text{number of edges} - \text{degree(reference vertex)})$. If every branch is assumed to contain one impedance element, each branch equation will contribute 1 entry for the branch current and 2 entries (or 1 if one of the vertices is the reference vertex) for node voltages; the total number of entries due to the branch equations will be $(3 \times \text{number of edges} - \text{degree(reference vertex)})$. The total non-zero entries in the MNA matrix will therefore be equal to $(5 \times \text{number of edges} - 2 \times \text{degree(reference vertex)})$. The number of non-zero entries in the MNA matrix.
matrix have been computed assuming the degree of the reference vertex to be 5. The number of non-zero entries in the loop impedance matrix and the MNA matrix for the same graphs have been given in Table 3. It can be seen that for graphs with nonplanarities less than 2%, the number of non-zero entries in the loop impedance matrix is even less than that in the case of MNA matrix, when the non-planarity is less than 2%. Since the time taken for solving the linear equations, at each iteration, of the Newton Raphson algorithm, is almost proportional to the number of non-zero entries in the matrix, the linear equation solution, at each iteration, is likely to be faster in case of loop analysis than MNA, for such graphs. For graphs with nonplanarity of around 10%, the number of non-zero entries for the loop impedance matrix is about twice that of the MNA. Even in these cases, the loop analysis method is expected to perform better than MNA for the complete analysis, if iterative methods (such as the Conjugate gradient) are used for the solution of linear equations. This is because, the number of iterations required in case of MNA, because of the poor structure of the coefficient matrix, appear to be two to three times those required in case of loop analysis.

4 Conclusions

Hybrid methods can compete favourably with MNA for most practical circuits if the loop matrix generation is done using the new method based on meshes, described in this paper. As this method is based on the property of planarity of the graph, the number of non-zero entries of the loop impedance matrix obtained here, increase with increase in nonplanarity of the graph. (Here, non-planarity refers to the number of additional edges in a graph over those of a complete planar graph on its vertices.) For graphs with nonplanarity less than 2%, the number of non-zero entries in the loop impedance matrix generated by using this method is even less than that in case of the MNA matrix. As such, in these cases, the linear equation solution at each iteration of the Newton Raphson algorithm is likely to be faster in case of loop analysis than with MNA. For graphs with nonplanarity of around 10%, the number of non-zero entries for the loop impedance matrix is about twice that of the MNA. Even in these cases, the loop analysis method is expected to perform better than MNA for the complete analysis, if iterative methods (such as the Conjugate gradient) are used for the solution of linear equations.

The computation time for loop construction is also very small compared to the total analysis time; a graph on 5000 vertices and 16494 edges took 8 minutes and 19 seconds (on the SUN 3/260 system) to form a loop matrix. The expected time of analysis of a dynamic circuit on such a graph is several hours.

References


5 Appendix

The outline of the complete algorithm for the method is as follows.

* procedure loop-mesh(G)
  */ This routine generates a loop matrix by forming meshes on planar slices of graph G */
  begin
    for each biconnected component Gj of G
      begin
        test planarity of Gj ;
        if Gj is nonplanar
          begin
            find meshes of planar subgraph Pj of Gj
            construct a graph Gz with the nonplanar subgraph Nj and a spanning tree of planar edges on vertices of the planar subgraph
          end
        else
          begin
            find meshes of Gj
            G2 = NULL ;
            end
            while (G2 not NULL) or (stack not empty)
              begin
                if (G2 not NULL)
                  begin
                    find biconnected components of G2;
                    push them on a stack ;
                    pop out one biconnected component G3 from the stack ;
                    end
                  else
                    pop out one biconnected component G3 from the stack ;
                    test planarity of G3 ;
                    if G3 is nonplanar
                      begin
                        find meshes of planar subgraph Pj
                      end
                end
            end
          end
of $G_3$;
+ planar
+ subgraph $N_3$ and a spanning tree of
planar
+ edges on vertices of the planar sub-
+ graph;
+ end
+ else
+ begin
    find meshes of $G_3$;
    $G_2$ = NULL;
+ end
+ end;

TABLE 1. Computation times for graphs with 10% nonplanarity.

<table>
<thead>
<tr>
<th>Vertices</th>
<th>Edges</th>
<th>Time in seconds</th>
</tr>
</thead>
<tbody>
<tr>
<td>250</td>
<td>819</td>
<td>8</td>
</tr>
<tr>
<td>1000</td>
<td>3294</td>
<td>46</td>
</tr>
<tr>
<td>2000</td>
<td>6594</td>
<td>133</td>
</tr>
<tr>
<td>5000</td>
<td>16494</td>
<td>499</td>
</tr>
<tr>
<td>10000</td>
<td>32994</td>
<td>1093</td>
</tr>
</tbody>
</table>

TABLE 2. Comparison of density of loop impedance matrices obtained with the fundamental circuit matrix method and the new loop matrix method for graphs on 1000 vertices with varying nonplanarity.

<table>
<thead>
<tr>
<th>Method</th>
<th>Fundamental circuit matrix (depth first search tree)</th>
<th>Fundamental circuit matrix (breadth first search tree)</th>
<th>Loop matrix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edges % non-planarity</td>
<td>Non-zero entries/ %density</td>
<td>Non-zero entries/ %density</td>
<td>Total entries</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>------------------------------------------------------</td>
<td>-------------------------------------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>2994 0.0</td>
<td>181725</td>
<td>100113</td>
<td>7977</td>
</tr>
<tr>
<td>3024 1.0</td>
<td>762248</td>
<td>152069</td>
<td>12601</td>
</tr>
<tr>
<td>3058 2.1</td>
<td>1180545</td>
<td>206811</td>
<td>15287</td>
</tr>
<tr>
<td>3144 5.0</td>
<td>2001069</td>
<td>314773</td>
<td>21687</td>
</tr>
<tr>
<td>3294 10.0</td>
<td>-</td>
<td>445863</td>
<td>31533</td>
</tr>
</tbody>
</table>

TABLE 3. Comparison of the non-zero entries in the loop impedance matrix obtained using the new loop matrix generation method and those in the MNA matrix.

<table>
<thead>
<tr>
<th>Vertices</th>
<th>Edges</th>
<th>% non-planarity</th>
<th>Total entries in loop impedance matrix</th>
<th>Nonzero entries in loop impedance matrix</th>
<th>Nonzero Entries in MNA matrix</th>
</tr>
</thead>
<tbody>
<tr>
<td>500</td>
<td>1494</td>
<td>0.0</td>
<td>990025</td>
<td>3997</td>
<td>7470</td>
</tr>
<tr>
<td>500</td>
<td>1509</td>
<td>1.0</td>
<td>1020100</td>
<td>6172</td>
<td>7535</td>
</tr>
<tr>
<td>500</td>
<td>1531</td>
<td>2.5</td>
<td>1065024</td>
<td>7594</td>
<td>7646</td>
</tr>
<tr>
<td>500</td>
<td>1569</td>
<td>5.0</td>
<td>1144990</td>
<td>9774</td>
<td>7835</td>
</tr>
<tr>
<td>500</td>
<td>1644</td>
<td>10.0</td>
<td>1311025</td>
<td>14197</td>
<td>8210</td>
</tr>
<tr>
<td>1000</td>
<td>2994</td>
<td>0.0</td>
<td>3980025</td>
<td>7977</td>
<td>14960</td>
</tr>
<tr>
<td>1000</td>
<td>3024</td>
<td>1.0</td>
<td>4100625</td>
<td>12601</td>
<td>15110</td>
</tr>
<tr>
<td>1000</td>
<td>3058</td>
<td>2.1</td>
<td>4239481</td>
<td>15287</td>
<td>15280</td>
</tr>
<tr>
<td>1000</td>
<td>3144</td>
<td>5.0</td>
<td>4601025</td>
<td>21687</td>
<td>15710</td>
</tr>
<tr>
<td>1000</td>
<td>3294</td>
<td>10.0</td>
<td>5267025</td>
<td>31533</td>
<td>16460</td>
</tr>
<tr>
<td>4000</td>
<td>11994</td>
<td>0.0</td>
<td>66902025</td>
<td>31877</td>
<td>69970</td>
</tr>
<tr>
<td>4000</td>
<td>12114</td>
<td>1.0</td>
<td>5853225</td>
<td>52907</td>
<td>60560</td>
</tr>
<tr>
<td>4000</td>
<td>12194</td>
<td>1.7</td>
<td>67158025</td>
<td>60715</td>
<td>60970</td>
</tr>
<tr>
<td>4000</td>
<td>12594</td>
<td>5.0</td>
<td>73874025</td>
<td>100373</td>
<td>62970</td>
</tr>
<tr>
<td>4000</td>
<td>13194</td>
<td>10.0</td>
<td>84548025</td>
<td>164461</td>
<td>65970</td>
</tr>
</tbody>
</table>

* Points of equal sparsity of the two matrices.