COMPUTING A SPARSE JACOBIAN MATRIX BY ROWS AND COLUMNS

A.K.M. SHAHADAT HOSSAIN and TROND STEIHAUG

Department of Informatics, University of Bergen, Høyteknologisenteret, N-5020 Bergen, Norway
Shahadat.Hossain@ii.uib.no, Trond.Steihaug@ii.uib.no

Efficient estimation of large sparse Jacobian matrices has been studied extensively in the last couple of years. It has been observed that the estimation of Jacobian matrix can be posed as a graph coloring problem. Elements of the matrix are estimated by taking divided difference in several directions corresponding to a group of structurally independent columns. Another possibility is to obtain the nonzero elements by means of the so called *Automatic differentiation*, which gives the estimates free of truncation error that one encounters in a divided difference scheme. In this paper we show that it is possible to exploit sparsity both in columns and rows by employing the forward and the reverse mode of Automatic differentiation. A graph-theoretic characterization of the problem is given.

KEY WORDS: AD, Forward and Reverse mode, Nonlinear Optimization, Numerical Differences, Sparsity.

1 INTRODUCTION

In many nonlinear optimization problems one often needs to estimate the Jacobian matrix of a nonlinear function $F : R^n \to R^m$. When the problem dimension is large and the underlying Jacobian matrix is sparse it is desirable to utilize the sparsity to improve the efficiency of the solutions to these problems.

Given the sparsity structure of a matrix A, we want to obtain vectors $d_1, d_2, \cdots, d_p$ so that the elements of the given matrix A are uniquely determined from the products $Ad_1, Ad_2, \cdots, Ad_p$ with p as small as possible. The problem can also be posed as to find vectors $d_1, d_2, \cdots, d_q$ so that the nonzero elements of A are uniquely determined from the products $A^T d_1, A^T d_2, \cdots, A^T d_q$ with q as small as possible. In the former we may take advantage of the known sparsity in the columns while in the latter sparsity in rows can be exploited. In this paper, we use *Automatic Differentiation (AD)* [8] to obtain estimates of the nonzero elements.

Automatic Differentiation is a chain-rule-based technique for evaluating the derivatives of functions defined by computer programs written in some high level language. The execution of a program computing the function

$$y = F(x), F : R^n \to R^m, \tag{1}$$

can be viewed as a sequence of scalar assignments

$$v_j = \phi_j(\{v_i : i \in \mathcal{P}_j\}). \tag{2}$$

Here the set $\mathcal{P}_j$ contains the indices of already computed quantities v_i. The *elementary* functions ϕ_j can be arithmetic operations or univariate transcendental functions.

If all these elementary functions ϕ_j are well defined and have continuous partial derivatives

$$c_{ji} \equiv \frac{\partial}{\partial v_i}\phi_j, i \in \mathcal{P}_j, \tag{3}$$

on some neighborhood

$$\mathcal{N}_j \subset R^{n_j}, n_j \equiv |\mathcal{P}_j|, \tag{4}$$

of their respective arguments $v_i, i \in \mathcal{P}_j$, then the nonzeros of the Jacobian matrix can be computed from the *elementary partials* c_{ji} defined above by the repeated application of the chain rule.

Let the variables v_j be numbered such that they can be combined into three vectors:

$$\begin{aligned} x &\equiv (v_1, v_2, \cdots, v_n) \\ z &\equiv (v_{n+1}, v_{n+2}, \cdots, v_{n+p}) \\ y &\equiv (v_{n+p+1}, v_{n+p+2}, \cdots, v_{n+p+m}), \end{aligned} \tag{5}$$

where x is the collection of independent variables, z is the collection of intermediate quantities and y is the collection of dependent variables. Suppose x is a differentiable function of some parameter. Denote the differentiation with respect to this parameter by the superscript prime. Then by chain rule for all $i > n$

$$v'_j = \sum_{i \in \mathcal{P}_j} c_{ji}.v'_i \in R. \tag{6}$$

If the initial tangent vector x' is initialized to the jth coordinate vector, the execution of above recurrence will yield

$$y' = F'(x)x' = F'_j(x), \tag{7}$$

the jth column of the Jacobian matrix at the current value of x. The above procedure of propagating the derivatives is known as the *forward mode* of the Automatic differentiation. Another technique that allows the computation of the rows of the Jacobian matrix is the *reverse mode* of the Automatic differentiation. Instead of propagating the elementary partial derivatives forward, in reverse mode, the *sensitivities* of the dependent variables with respect to intermediate quantities are propagated in the reverse direction. In the reverse mode the jth row of the Jacobian matrix can be obtained as the product $\hat{y}^T F'(x)$ when the vector $\hat{y}$ is initialized to jth coordinate vector. An excellent account of the recent developments in Automatic differentiation can be found in [5].

Much of the research on the efficient estimation of sparse Jacobian and Hessian matrices [1, 2, 4, 6, 7, 9, 10] uses divided differences to obtain estimates of the nonzeros. In this approach one forms groups of columns that are *structurally orthogonal* i.e., columns that do not have nonzero in the same row position. The estimates of the nonzeros in those columns are then obtained from a divided difference formula. Curtis, Powell and Reid [1] were the first to exploit the property of structural orthogonality in an effort to reduce the number of function evaluation. Coleman and Moré [2] showed that forming groups of structurally orthogonal columns can be seen as a *graph coloring* problem. Their observation led to several coloring heuristics with improved performance. Steihaug and Hossain [10] have suggested a technique where coloring heuristics from Coleman and Moré [2] are used on blocks of rows of the Jacobian matrix. This property of structural orthogonality can also be utilized when we use AD to obtain the nonzero elements. Here we initialize the vector x such that $x_j = 1$ for all the columns j that are structurally orthogonal and all other entries of x are set to 0 and use the forward mode to obtain the nonzeros. Alternatively, a set of structurally orthogonal rows can be estimated in reverse mode by properly initializing the vector $\hat{y}$. Thus one can use the forward mode to compute a group of columns from the matrix-vector product Ad where vector d has nonzeros in positions corresponding to columns of A that are structurally orthogonal. Also a group of structurally orthogonal rows can be estimated by vector-matrix product $d^T A$ in the reverse mode. This shows that advantage of the sparsity can be exploited both in columns and in rows [3].

In this paper we analyze the problem of the estimation of a Jacobian matrix from a graph theoretic view point. In particular, we show how the known sparsity structure can be exploited in computing rows and columns. Methods for computing the Jacobian matrix by partitioning rows and columns have been studied in [11, 12].

The main results of the paper is given in Section 2. We begin Section 2 by considering the partitioning of the rows and the columns. We give examples where either row or column partitioning alone may not be able to take full advantage of the known sparsity. A more general partitioning problem is defined where both rows and columns are grouped. The partitioning problem is then characterized as a restricted graph coloring problem of a bipartite graph. We then introduce the concept of complete direct cover. We show that row-column consistent partition is likely to reduce the number of groups.

In Section 3 we present an algorithm that can be used to obtain a path p-coloring of the bipartite graph associated with the matrix A.

Numerical results are given in Section 4. We use the matrices from Harwell collection to test the path p-coloring algorithm.

Finally, some concluding remarks are given in Section 5.

2 PARTITION PROBLEM AND COLORING

Automatic differentiation can be used to compute the nonzeros of the Jacobian matrix. By partitioning the columns or the rows of the matrix, the nonzeros can be computed using the forward or the reverse mode respectively.

A *partition* of the columns and rows of A is the division of columns and rows into groups such that each column and row belongs to exactly one group and each group contains either columns or rows.

A partition of the columns and rows such that for each nonzero element a_{ij} in the matrix, columns (rows) that are in the same group of column j (row i) do not have nonzero in row i (column j) is said to be *consistent* with the direct determination of A.

To see that the partitioning either columns or rows leads to a direct determination of A consider a group of columns C. Define a vector d which has component $\delta_j \neq 0$ if column j is in the group and $\delta_j = 0$ if column j is not included in the group. Then

$$Ad = \sum_{j \in C} \delta_j a_j, \tag{8}$$

where a_j, $j = 1, 2, \cdots, n$ are the columns of A. If the group does not contain more than one column that have nonzeros in the same row position then

$$(Ad)_i = \delta_j a_{ij}, \tag{9}$$

where column $j \in C$ has a nonzero in row i. Analogously, nonzeros can also be obtained from a group of rows.

The above discussion shows that we can take advantage of the known sparsity in the Jacobian matrix while using Automatic differentiation techniques to obtain the estimate of the nonzeros. Much of the work done on the sparse Jacobian matrix estimation based on finite differences [1, 2] partitions the columns of the matrix into groups of columns. This way, sparsity can only be exploited in columns. In contrast, Automatic differentiation technique allows us to use the sparsity both in columns and in rows.

The following example illustrates that it might be advantageous to exploit the sparsity in rows.

$$\begin{pmatrix} \times & & & & \\ & \times & & & \\ & & \times & & \\ & & & \times & \\ \times & \times & \times & \times & \times \end{pmatrix}$$

FIGURE 1: Sparsity pattern of a matrix

A column partition of the matrix in Fig. 1 requires at least 5 groups as the last row consists entirely of nonzeros. But if we consider row partition only 2 groups are needed.

This example suggests taking the column and row partition separately and use the one which yields the minimum number of groups. However, this approach might not be satisfactory with the matrices with few dense rows and columns. Consider the matrix in Fig. 2.

$$\begin{pmatrix} \times & & & & \times \\ & \times & & & \times \\ & & \times & & \times \\ & & & \times & \times \\ \times & \times & \times & \times & \times \end{pmatrix}$$

FIGURE 2: Sparsity pattern of a matrix

Both the row and column partition would require at least 5 groups as the row 5 and column 5 are dense. One possible remedy to this problem is to evaluate row 5 and column 5 separately and for rest of the nonzeros of the matrix we can collect all the remaining columns (or rows) in one group. Thus we can estimate the nonzeros in the matrix with only 3 groups. From these examples it appears that it might be worthwhile to consider the partition problem where both rows and columns are combined into groups such that the nonzeros can be estimated uniquely. We can pose the partition problem in the following way.

Given a matrix $A \in R^{m\times n}$ find the matrices $D_1 \in R^{n\times p_1}$ and $D_2 \in R^{m\times p_2}$ such that $D_2^T A$ and AD_1 determines the matrix A uniquely with $p = p_1 + p_2$ groups.

DEFINITION 2.1. A partition of the rows and columns of the matrix A is *row-column consistent* if each group consists entirely of either rows or columns and for every nonzero a_{ij} of A either column a_j is in a group where no other column in its group has a nonzero in row r_i or row r_i is in a group where no other row in its group has a nonzero in column a_j.

It is clear that given a row-column consistent partition we can obtain each nonzero element a_{ij} of A either from a column group where the only column containing a nonzero in row r_i is column a_j or from a row group where the only row containing a nonzero in column a_j is row r_i.

The above characterization of estimating the Jacobian matrix naturally leads to the following formulation.

Row-column consistent partition problem: Obtain a row-column consistent partition such that the number of groups $p = p_1 + p_2$ is minimized.

This is a combinatorial problem and many such problems have been successfully analyzed and a better insight into the problem has been obtained by connecting them with graph theoretic problems. Our aim is to give a graph theoretic characterization of the row-column consistent partition problem.

A *graph* G is an ordered pair (V, E) where elements of $V \neq \emptyset$ are called *vertices* and the elements of E are unordered pairs of distinct vertices which are called *edges*. Two vertices u and v are *adjacent* if (u, v) is an edge with *end points* u and v. In this paper we consider graphs that have finite number of vertices and that do not have *multiple edges* i.e., there is only one edge between any two distinct vertices.

A $p-coloring$ of a graph is a function

$$\phi : V \to \{1, 2, \cdots, p\}, \tag{10}$$

such that $\phi(u) \neq \phi(v)$ if vertices u and v are adjacent. The *chromatic number* $\chi(G)$ is the smallest p for which G has a p-coloring. A coloring ϕ of G *induces* a partition of the vertices in V into groups such that

$$C_k = \{u \in V : \phi(u) = k\}, \tag{11}$$

A graph G_b is *bipartite* if its vertices V can be divided into two disjoint sets V_1 and V_2 in such a way that every edge of G has a vertex from V_1 and a vertex from V_2 as end points.

A *path* P in G of *length* l is a sequence $\{v_1, v_2, \cdots, v_{l+1}\}$ of distinct vertices in G such that v_i is adjacent to v_{i+1}, for $1 \leq i \leq l$.

Given a matrix $A \in R^{m\times n}$ we define a bipartite graph $G_b(A)$ which has two sets of vertices $V_1 = \{a_1, a_2, \cdots, a_n\}$ and $V_2 = \{r_1, r_2, \cdots, r_m\}$ where a_j is the jth column and r_i is the ith row of A. There is an edge $(r_i, a_j) \in E$ whenever a_{ij} is a nonzero element of A.

DEFINITION 2.2. Let G_b be a bipartite graph with vertices partitioned into two disjoint sets V_1 and V_2. A coloring ϕ is *path p-coloring* of G_b if every path in G_b of length 3 uses at least 3 colors and

$$\{\phi(v_1) : v_1 \in V_1\} \cap \{\phi(v_2) : v_2 \in V_2\} = \emptyset, \tag{12}$$

The smallest p for which the graph G_b is path p-colorable is denoted by $\chi_p(G_b)$.

THEOREM 2.3. *Let A be a $m \times n$ matrix. The mapping ϕ induces a row-column consistent partition of the rows and columns of A if and only if ϕ is a path p-coloring of $G_b(A)$.*

Proof. Assume that ϕ is a path p-coloring of A. Since ϕ is a coloring it induces a partition of the rows and columns of A. Now if this is not a row-column consistent partition then there is a nonzero a_{ij} in the matrix and columns a_q, $q \neq j$ and rows r_p, $p \neq i$ such that columns a_q and a_j are in the same group with $a_{iq} \neq 0$ and rows r_i and r_p are in the same group with $a_{pj} \neq 0$. Then $\phi(a_j) = \phi(a_q)$ and $\phi(r_i) = \phi(r_p)$. Hence ϕ is a 2-coloring of the path

$$P = (r_p, a_j, r_i, a_q). \tag{13}$$

Since ϕ is a path p-coloring of $G_b(A)$, it implies that P is a path of length $l < 3$. Hence

$$\phi(r_p) = \phi(r_i) = \phi(a_j) = \phi(a_q), \tag{14}$$

which is contradicts (12). Thus the partition is row-column consistent.

To prove the converse, assume that ϕ induces a row-column consistent partition of the rows and columns of A. Consider nonzero a_{ij}. By definition row r_i and column a_j are in different groups and hence ϕ is a coloring of $G(A)$. To show that it is a path p-coloring let

$$P = (r_p, a_j, r_i, a_q), \tag{15}$$

be a path of length $l = 3$. Then a_{ij}, a_{pj} and a_{iq} are nonzeros in the matrix. If row r_i is the only row in its group with a nonzero in column a_j then $\phi(r_i) \neq \phi(r_p)$. Similarly, if column a_j is the only column in its group with a nonzero in row r_i then $\phi(a_j) \neq \phi(a_q)$. Hence ϕ uses at least 3 colors for the path P. This completes the proof. □

Theorem 2.3 establishes the desired connection between the problem of estimation of a sparse Jacobian matrix by partitioning the rows and columns and the restricted coloring of a graph.

An important observation to be made here is that in some instances one or more groups (color) are redundant in the sense that all the nonzeros are already determined by the other groups (colors). This fact is illustrated by the following example.

$$\begin{pmatrix} & & \times \\ & & \times \\ \times & \times & \times \end{pmatrix}$$

FIGURE 3: Sparsity pattern of a matrix

A row-column consistent partition for this matrix may have column a_3 in one group (group 1), rows r_3, r_2 and r_1 in another group (group 2) and rest of the columns in one separate group (group 3). Nonzeros a_{13}, a_{23} and a_{33} are determined by group 1 and nonzeros a_{31} and a_{32} are determined by group 2. Hence, group 3 is not needed at all. This example shows that it is desirable to focus on finding the groups of columns and rows that together completely determine the nonzeros of the matrix.

McCormic [9] introduces the notion of direct cover property in connection with the approximation of sparse Hessian matrices. We define a similar concept regarding Jacobian matrix estimation.

Let X be a group of columns or rows in a row-column partition.

DEFINITION 2.4. A nonzero a_{ij} such that column $a_j \in X$ or row $r_i \in X$ is said to be *covered* by X.

DEFINITION 2.5. A nonzero a_{ij} covered by X is said to be *directly determined* by X if there are no column a_q (row r_p) in X that has a nonzero in row r_i (column a_j).

DEFINITION 2.6. Let S_c be a collection of subsets of columns and S_r be a collection of subsets of rows. The set $\{S_c, S_r\}$ is called *complete direct cover* of A if

- The intersection of any two subsets is empty
- For each nonzero element a_{ij}, there is a subset $X \in S_c \cup S_r$ such that a_{ij} is directly determined by X.

The above definition of complete direct cover gives way for columns or rows not necessarily be included in any subset of rows or columns as long as all the nonzeros are determined directly. An interesting question in this regard is how the cardinality of a row-column consistent partition and a complete direct cover is related. An algorithm for computing complete direct cover amounts to finding groups of rows and columns that satisfies the direct cover property. We can terminate the algorithm when all the nonzeros in the matrix are determined. The resulting groups of rows and columns constitutes a complete direct cover. Now consider the rows and columns that are not included in the direct cover. Let us collect all the columns that are left out in the group X_c and all the rows that are left out in the group X_r. We claim that these two groups together with the groups of rows and columns in the complete direct cover constitutes a row-column consistent partition. In other words, we have a valid path p-coloring of the bipartite graph where each group of column and each group of row receives a color. Because of the direct cover property any path of length 3 involving rows and columns that are included in the groups other than X_c and X_r uses 3 colors. Also there cannot be any edge between columns in X_c and rows in X_r. Thus the only path we need to consider are the paths of the form (a_1, r_1, a_2, r_2) where $a_1 \in X_c$ and $r_1 \notin X_r$ or (r_1, a_1, r_2, a_2) where $r_1 \in X_r$ and $a_1 \notin X_c$. But these paths use at least 3 colors. Hence this is a valid path p-coloring. This discussion is summarized in the following result.

LEMMA 2.7. *Given a complete direct cover (S_c, S_r) for the matrix A, a row-column consistent partition of the rows and columns of A can be constructed that contains at most $|S_c| + |S_r| + 2$ groups.*

The above bound on the number of colors is tight as the following example illustrates.

$$\begin{pmatrix} \times & & \times & \\ & \times & & \\ & \times & & \times \\ & & \times & \end{pmatrix}$$

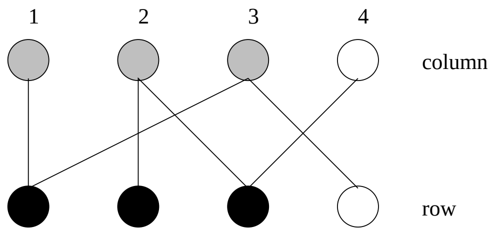

FIGURE 4: A matrix and the corresponding bipartite graph

In Fig. 4, a complete direct cover is obtained by grouping columns 1, 2 and 3 in one group and rows 1, 2 and 3 in another group; column 4 and row 4 are left out. This direct cover is optimal. A path p-coloring from this direct cover needs two extra color: one each for column 4 and row 4 requiring a total of 4 colors. On the other hand, we can construct a direct cover with columns 2 and 3 in one group and columns 1 and 4 in another group. A path p-coloring is then defined by using one extra color on rows requiring only 3 colors.

We have seen examples where a row-column partition (and equivalently path p-coloring) can be used to reduce the number of groups (colors). It is useful to know whether this is true in general.

The *column intersection graph* $G(A)$ associated with A is a graph with vertex set $\{a_1, a_2, \cdots, a_n\}$ where a_j is the jth column of A and edge (a_j, a_q) if $j \neq q$ and columns a_j and a_q have a nonzero in the same row position.

The column intersection graph associated with A^T is known as the *row intersection graph* $G(A^T)$.

From the definition of p-coloring and column intersection graph it follows that [2] ϕ is a p-coloring of $G(A)$ $(G(A^T))$ if and only if ϕ induces a consistent partition of the columns (rows) of A.

We can now state the connection between general p-coloring of $G(A)$ and $G(A^T)$ and the path p-coloring of $G_b(A)$.

THEOREM 2.8. $\chi_p(G_b(A)) \leq min\ (\chi(G(A^T)), \chi(G(A))) + 1$

Proof. Without loss of generality let us assume that

$$\chi(G(A)) = min(\chi(G(A^T)), \chi(G(A))).$$

We show that with minor change a coloring ϕ of the column intersection graph $G(A)$ is also a path p-coloring of $G_b(A)$. First we color the vertices in $G_b(A)$ that correspond to rows in the matrix A with one color that would not be used further. Let $P = (r_p, a_j, r_i, a_q)$ be any path of length 3 in $G_b(A)$. Then the elements a_{ij} and a_{iq} are nonzeros. Then the columns a_j and a_q have different color under the coloring ϕ. This implies that the path P uses at least 3 colors. Hence, ϕ does not violate the path p-coloring condition. This establishes the above inequality. □

It is important to note that in the proof the extra color used on the vertices corresponding to rows can be dispensed with. In fact, coloring of $G(A)$ $(G(A^T))$ induces a complete direct cover. Theorem 2.8 illustrates that the row-column partitioning is superior to simple partitioning in that the number of groups in a complete direct cover may be smaller.

In view of Lemma 2.7, a path p-coloring can be constructed from a given complete direct cover with no more than 2 additional colors. On the other hand it is not obvious how to detect the redundant groups of columns and rows in a path p-coloring. However, our main concern is the computation of the Jacobian matrix efficiently and we give a simple direct cover algorithm in the next section.

3 AN ALGORITHM FOR COMPLETE DIRECT COVER

In this section we discuss an algorithm that can be used to find a complete direct cover of A. The algorithm that we propose here utilizes the connection between the path p-coloring of $G_b(A)$ and the complete direct cover of A. We find it convenient to discuss the algorithm in graph theoretic terms.

Although there exists many coloring algorithms (see [2]) for general graphs, they are not directly applicable to the graph $G_b(A)$. The algorithm we consider here is based on the work of Powell and Toint [6].

We give an algorithm for path p-coloring that is applicable to bipartite graphs. We have to take into account the fact that in a row-column consistent partition we cannot put rows and columns in the same group. In terms of graph, the restriction is that two sets of vertices that define the bipartite graph could not be mixed. To define the algorithm we need some graph theory definitions.

Given a graph $G = (V, E)$ the *degree* of a vertex v is the number of edges with v as an endpoint. Given a nonempty subset $V_0 \subset V$ the *subgraph* $G[V_0] = (V_0, E_0)$ is *induced* by V_0 if $E_0 = \{(u, v) \in E : u, v \in V_0\}$. We also say that a group of vertices $v_i, i = 1, \cdots, k$ are in the *same bi-partition* if either $v_i \in V_1, i = 1, \cdots, k$ or $v_i \in V_2, i = 1, \cdots, k$.

Algorithm:

Let $G_b = (V_1 \cup V_2, E)$ be a bipartite graph
$k := 0$
edgecount := 0
<u>while</u> edgecount $< |E|$ <u>do</u>
  $k := k + 1$
  Let $V_1^k \subset V_1$ and $V_2^k \subset V_2$ be the uncolored vertices
  Sort the vertices of $G[V_1^k \cup V_2^k]$ in nonincreasing degree in $G[V_1^k \cup V_2^k]$
  Let the order be defined by $w_i, i = 1, 2, \cdots, |V_1^k \cup V_2^k|$
  $W^k := \{w_1\}$
  <u>for</u> $i = 2, 3, \cdots, |V_1^k \cup V_2^k|$ <u>do</u>
    <u>if</u> w_i and w_1 are in the same bi-partition <u>and</u>
    there is no $w \in W^k$ such that w_i is connected to w by a path in
    $G[V_1^k \cup V_2^k]$ of length $l = 2$ <u>then</u>
      Add w_i to the set W^k
    <u>endif</u>
  <u>endfor</u>
  Assign the color k to every vertex in W^k
  Let E_k be the set of edges which has end points in W^k
  edgecount := edgecount $+|E_k|$
<u>endwhile</u>

The above algorithm processes the rows and columns until all the edges are accounted for. Note that en edge in $G_b(A)$ corresponds to a nonzero in the matrix A. A complete direct cover for the matrix A is computed by the *while* loop. Lemma 2.7 shows that uncolored rows and columns can be included in two additional groups. We can put instructions after *while* loop that will collect all those uncolored columns and rows and assign at most two extra colors to give a path p-coloring.

In the algorithm, vertices are selected in nonincreasing degree in the induced subgraph. Other ordering of the vertices such as incidence degree ordering [2] is also possible.

4 NUMERICAL RESULTS

We implemented the algorithm for finding complete direct cover that is described in this paper. The test matrices comprised of Harwell test collection. In the implementation of the algorithm ties in the degree of vertices are broken arbitrarily. We also report test results obtained by applying dsm [4] on the test matrices (both A and A^T). The following statistics are generated.

TABLE 1: Harwell Matrices (Static Information)

Matrix	n	m	nnz	$dnsm$	mxr	mnr	mng	mxc	mnc	mng'
abb313	176	313	1557	2.83	6	1	10	26	2	26
ash219	85	219	438	2.35	2	2	3	9	2	9
ash292	292	292	1250	1.47	8	1	8	10	1	10
ash331	104	331	662	1.92	2	2	6	12	3	12
ash608	188	608	1216	1.06	2	2	5	12	2	12
ash958	292	958	1916	0.68	2	2	6	13	3	13
curtis54	54	54	291	9.98	12	3	12	16	3	16
ibm32	32	32	126	12.30	8	2	8	7	2	7
will199	199	199	701	1.77	6	1	7	9	2	9
will57	57	57	281	8.65	11	2	11	11	2	11
will701	199	199	701	1.77	6	1	7	9	2	9
gent113	113	113	655	5.13	20	1	20	27	1	27
arc130	130	130	1282	7.59	124	1	124	124	1	124
shl	663	663	1687	0.38	422	1	422	4	1	4
shl200	663	663	1726	0.39	440	1	440	4	1	4
shl400	663	663	1712	0.39	426	1	426	4	1	4
str	363	363	2454	1.86	34	1	34	34	1	34
str200	363	363	3068	2.33	30	1	30	26	1	26
str400	363	363	3157	2.40	33	1	33	34	1	36
str600	363	363	3279	2.49	33	1	33	34	1	36
bp	822	822	3276	0.48	266	1	266	20	1	20
bp200	822	822	3802	0.56	283	1	283	21	1	21
bp400	822	822	4028	0.60	295	1	295	21	1	21
bp600	822	822	4172	0.62	302	1	302	21	1	21
bp800	822	822	4534	0.67	304	1	304	21	1	21
bp1000	822	822	4661	0.69	308	1	308	21	1	21
bp1200	822	822	4726	0.70	311	1	311	21	1	21
bp1400	822	822	4790	0.71	311	1	311	21	1	21
bp1600	822	822	4841	0.72	304	1	304	21	1	21
fs541-1	541	541	4285	1.46	11	1	11	541	5	541
fs541-2	541	541	4285	1.46	11	1	11	541	5	541
eris	1176	1176	9864	0.71	81	1	81	93	1	93
lundA	147	147	1298	6.01	12	1	12	12	1	12
lubdB	147	147	1294	5.99	12	1	12	12	1	12

n - Number of columns in A
m - Number of rows in A
nnz - Number of nonzeros in A
mxr - Maximum number of nonzeros in any row of A
mxc - Maximum number of nonzeros in any column of A
mnr - Minimum number of nonzeros in any row of A
mnc - Minimum number of nonzeros in any column of A
mng - A lower bound on the number of column groups
mng' - A lower bound on the number of row groups

dnsm - matrix density (percentage)

TABLE 2: Harwell Matrices (Coloring information)

	dsm		*Complete direct cover*		
Matrix	*mxg*	*mxg'*	*cg*	*rg*	*tg*
abb313	10	26	13(0)	0(1)	13(1)
ash219	4	9	5(0)	0(1)	5(1)
ash292	8	10	3(1)	7(0)	10(1)
ash331	6	12	6(0)	0(1)	6(1)
ash608	6	12	7(0)	0(1)	7(1)
ash958	6	13	6(0)	0(1)	6(1)
curtis54	12	16	1(1)	9(0)	10(1)
ibm32	8	7	8(0)	0(1)	8(1)
will199	7	9	6(0)	1(1)	7(1)
will57	11	11	2(1)	7(0)	9(1)
will701	7	9	6(0)	1(1)	7(1)
gent113	20	27	10(1)	8(0)	18(1)
arc130	124	124	16(1)	10(0)	26(1)
shl	422	4	0(1)	4(0)	4(1)
shl200	440	4	0(1)	4(0)	4(1)
shl400	426	4	0(1)	4(0)	4(1)
str	34	35	18(0)	6(1)	24(1)
str200	30	31	0(1)	31(0)	31(1)
str400	33	36	1(1)	35(0)	36(1)
str600	33	36	26(0)	9(1)	35(1)
bp	266	20	2(1)	14(0)	16(1)
bp200	283	21	6(1)	12(0)	18(1)
bp400	295	21	7(1)	12(0)	19(1)
bp600	302	21	7(0)	11(1)	18(1)
bp800	304	21	0(1)	21(0)	21(1)
bp1000	308	21	0(1)	22(0)	22(1)
bp1200	311	21	0(1)	22(0)	22(1)
bp1400	311	21	0(1)	22(0)	22(1)
bp1600	304	21	0(1)	21(0)	21(1)
fs541-1	13	541	15(0)	0(1)	15(1)
fs541-2	13	541	15(0)	0(1)	15(1)
eris	81	93	10(1)	70(0)	80(1)
lundA	12	13	0(1)	14(0)	14(1)
lubdB	12	13	0(1)	14(0)	14(1)
Total	4462	1824	196 (21)	391 (13)	587 (34)

mxg - Number of groups found by dsm on A
mxg' - Number of groups found by dsm on A^T
rg - Number of row groups in the complete direct cover
cg - Number of column groups in the complete direct cover
tg - rg + cg

Table 2 summarizes the experimental results. The numbers given in parenthesis in the columns 3 and 4 are the number of redundant column and row groups respectively. Total number of redundant groups i.e., sum of redundant column groups and redundant row groups are given in parenthesis in the column 5. The redundant groups are not counted in the row and column groups in the computed direct cover.

When the direct cover algorithm terminates, columns and rows without a group assignment can be given new group number so that at most two extra groups are needed. This way we get a path p-coloring of the bipartite graph from the complete direct cover.

We notice that for the test matrices considered the algorithm produced exactly one redundant group.

The performance of the path p-coloring algorithm is quite satisfactory on the test matrices as compared to the *dsm*. Out of 34 test matrices the path p-coloring algorithm performs at least as good as *dsm* on 23 of them. Total number of colors required by *dsm* and path p-coloring is 4462 and 587 respectively. This is approximately 7.5 fold reduction in the number colors. One of the reasons for this dramatic reduction is due to the fact that the number of nonzeros in a row gives a lower bound for the number of colors. This fact can be verified by comparing *maxr*, *maxg* and *tg* in the table 1.

The above observation is also evident in the number of colors required by dsm when applied on the transpose of the test matrices. Total number of colors required by path p-coloring (587) compares quite favorably with the total number of colors required by dsm (1824).

5 CONCLUDING REMARKS

We have attempted to exploit the sparsity structure in a Jacobian matrix in terms of both rows and columns in determining the nonzeros of the matrix. The forward mode and the reverse mode of Automatic differentiation can be employed to our advantage to obtain the nonzeros. It has been shown that the problem can also be posed as a graph coloring problem. Our results have shown that, for the matrices considered, a combination of row and column computation may result in fewer number of colors. Although, our results show significant reduction in the number of groups it will only be useful if we have efficient and user-friendly Automatic differentiation tools readily available. Another concern is the difference between forward and reverse mode in terms of temporal and spatial complexity. In general, the reverse mode requires more space than the forward mode as it has to store the computational graph of the function. These issues need to be investigated further.

REFERENCES

1. Curtis, A. R. , Powell, M. J. D. and Reid J. K. (1974), *On the estimation of sparse Jacobian matrices,* J. Inst. Math. Appl., 13 , pp. 117–119.
2. Coleman, T. F. and Moré J. J. (1983), *Estimation of sparse Jacobian matrices and graph coloring problems,* SIAM J. Numer. Anal., 20 , pp. 187–209.
3. Griewank, A.(1995), Personal communication.
4. Coleman, T. F., Garbow, B. and Moré, J. J.(1984) , *Software for estimating sparse Jacobian matrices,* ACM Trans. Mathematical Software, 10 , pp. 329–347.
5. Griewank, A. and Corliss, G. F.(1991), eds., *Automatic Differentiation of Algorithms: Theory, Implementation, and Application,* SIAM, Philadelphia.

6. Powell, M. J. D. and Toint, Ph. L.(1979), *On the estimation of sparse Hessian matrices,* SIAM J. Numer. Anal., 16, pp. 1060-1074.
7. Coleman, T. F. and Moré, J. J.(1984), *Estimation of sparse Hessian matrices and graph coloring problems,* Math. Programming, 28 , pp. 243–270.
8. Rall, L. B.(1981), *Automatic differentiation: Techniques and Applications*, vol. 120 of Lecture Notes in Computer Science, Springer-Verlag, Berlin.
9. McCormick, S. T.(1983), *Optimal approximation of sparse Hessians and its equivalence to a graph coloring problem*, Math. Programming, 26, pp. 153-171.
10. Steihaug, Trond and Hossain, A. K. M. S. (1992), *Graph coloring and the estimation of sparse Jacobian matrices using row and column partitioning,* Report 72, Department of Informatics, University of Bergen.
11. Steihaug, Trond and Hossain, A. K. M. S. , *Computing a Sparse Jacobian Matrix by Rows and Columns*, Report 109, Department of Informatics, University of Bergen, June 1995.
12. Coleman, T. F. and Verma, A., *The Efficient Computation of Sparse Jacobian Matrices Using Automatic Differentiation*, Tech. Report CTC95TR225, Cornell Theory Center, Cornell University, November 1995.