Bull. Korean Math. Soc. **41** (2004), No. 1, pp. 125–135

RECOGNITION OF STRONGLY CONNECTED COMPONENTS BY THE LOCATION OF NONZERO ELEMENTS OCCURRING IN $C(G) = (D - A(G))^{-1}$

KOONCHAN KIM* AND YOUNGYUG KANG

ABSTRACT. One of the intriguing and fundamental algorithmic graph problems is the computation of the strongly connected components of a directed graph G. In this paper we first introduce a simple procedure for determining the location of the nonzero elements occurring in B^{-1} without fully inverting B, where $B \equiv (b_{ij})$ and B^T are diagonally dominant matrices with $b_{ii} > 0$ for all i and $b_{ij} \leq 0$, for $i \neq j$, and then, as an application, show that all of the strongly connected components of a directed graph G can be recognized by the location of the nonzero elements occurring in the matrix $C(G) = (D - A(G))^{-1}$. Here $A(G)$ is an adjacency matrix of G and D is an arbitrary scalar matrix such that $(D - A(G))$ becomes a diagonally dominant matrix.

1. Introduction

Two vertices u and v in a directed graph $G = (V, E)$ without multiple arcs are *strongly connected*(denoted by $u \leftrightarrow v$) if there is a directed path from u to v and a directed path from v to u. A *strongly connected component*(SCC) of G is a maximal vertex set where all pairs of vertices are strongly connected. The set of all of the SCCs forms a partition of the vertex set V and its cardinality, the total number of SCCs, is denoted by $\beta_o(G)$.

Algorithms and methods have been proposed and developed for computing the SCCs of a directed graph G. Tarjan [8] proposed a linear-time

---

Received July 14, 2003.

2000 Mathematics Subject Classification: 05C20, 05C50, 65F05.

Key words and phrases: strongly connected components, directed graph, inverse matrix, diagonally dominant matrix.

*The present research has been conducted by the Attached Research Institutes Research Grant provided by the Office of Research Affairs of Keimyung University in 1998.

algorithm($O(n, m)$) and it is based on the depth-first search tree and the assignment of the 'lowpoint' value to each of the vertices in G. Sharir [7] studied an algorithm that requires two passes over the graph, and Gabow [4] presented one-pass algorithm that only maintains a representation of the depth-first search path. The concept of reachability matrix $R(x_i)$, which is defined as the union of the sets of vertices which are reachable from x_i, with Boolean operations on the adjacency matrix is used for computing the SCCs in [2, 6]. Recently, Wegener [10] has shown a more simple correctness proof of one of these algorithms described in [1, 3].

The purpose of this paper is first to present a simple technique that determines the location of the nonzero elements occurring in B^{-1}, where $B \equiv (b_{ij})$ and B^T are diagonally dominant matrices with $b_{ii} > 0$ for all i and $b_{ij} \leq 0$, for $i \neq j$, without fully inverting B and second to show that all of the SCCs of a directed graph G can also be recognized(computed) by the location of the nonzero elements occurring in $C(G) = (D - A(G))^{-1}$, where $A(G)$ is the usual adjacency matrix of the directed graph G, $D = dI$ is a scalar matrix, and I is the identity matrix. The diagonal element d is chosen so that $(D - A(G))$ becomes a diagonally dominant matrix and $C(G)$ can be expressed as an infinite sum of the matrix powers of $A(G)$.

2. A procedure for determining the location of nonzero elements

Suppose we are given an $n \times n$ diagonally dominant matrix $B \equiv (b_{ij})$ with $b_{ii} > 0$ for all i and $b_{ij} \leq 0$ for $i \neq j$. We suppose B^T is also a diagonally dominant matrix, where T denotes the transposition. That is, $b_{ii} > \sum_{j=1, j\neq i}^{n} |b_{ij}|$ for all i and $b_{jj} > \sum_{i=1, i\neq j}^{n} |b_{ij}|$ for all j.

Since B and B^T are diagonally dominant and $b_{ii} > 0$ for all i, B is a positive definite matrix(Golub [5, p.7]). This implies that B can be decomposed into $B = LU$, where L is a unit lower triangular matrix and U is an upper triangular matrix with $u_{ii} > 0$ for all i. See Golub [5, p.86].

One approach for finding B^{-1} is to apply a sequence of elementary row operations(EROs) on the augmented matrix $[B \mid I]$ and to transform it into the form $[I \mid B^{-1}]$. The process of reducing B to I consists of two steps: namely, 'forward elimination step' (reducing B to an upper triangular matrix U and at the same time transforming I to a lower triangular matrix L) and 'backward elimination step'(reducing U to I and transforming L to B^{-1}).

When applying EROs on the augmented matrix $[B \mid I]$, the following properties can be observed:

1. Let LU be a decomposition of B. Then $u_{ii} > 0$ for all i, as discussed above. This implies that the pivot elements i.e. the diagonal elements of B during the forward elimination step are always positive. Note that U obtained from LU decomposition is the same as the upper triangular matrix yielded when the forward elimination step has been completed on this augmented matrix.
2. Since we start with $b_{ij} \leq 0$ for all nondiagonal elements and since all the pivot elements are positive, the following properties can be observed during forward and backward elimination steps:

   (1) In eliminating a nonzero(negative) element, say $b_{ji}, j > i$, below the diagonal element b_{ii}, $-b_{ji}/b_{ii} > 0$ is multiplied to the ith row and its result is added to the jth row. Let $b_{ik} < 0, k > i$ be any nonzero element in the ith row. If $b_{jk} = 0$, then the new value at the jkth position becomes negative and if $b_{jk} < 0$, then the new value at this jkth position increases negatively since $-b_{ji}b_{ik}/b_{ii} < 0$. This implies that whether newly generated or not nonzero nondiagonal elements of B become always negative. At the same time, $-b_{ji}/b_{ii} > 0$ is added to the jith position in the matrix I, the right-hand side of the augmented matrix. Similarly, it can be seen that nonzero nondiagonal elements below the diagonal of I become always positive.

   (2) At the end of the forward elimination step, B turns into an upper triangular matrix $U = (u_{ij})$ in which the diagonal elements are all positive and nonzero nondiagonal elements(above diagonal) are all negative and I turns into a lower triangular matrix $L = (l_{ij})$ in which the diagonal elements are all equal to 1 and nonzero nondiagonal elements(below diagonal) are all positive, i.e. the augmented matrix turns into the form $[U \mid L]$.

   (3) In eliminating a nonzero(negative) element, say $u_{ji}, j < i$, above the diagonal element u_{ii}, $-u_{ji}/u_{ii} > 0$ is multiplied to the ith row and its result is added to the jth row. This implies that $-u_{ji}/u_{ii} > 0$ is added to the jith position in L. Let $l_{ik} > 0, k < i$ be any nonzero element in the ith row. If $l_{jk} = 0$, then the new value at the jkth position becomes positive and if $l_{jk} > 0$, then the new value at this jkth position increases positively since $l_{ji}l_{ik}/l_{ii} > 0$. This implies that diagonal elements and nonzero nondiagonal

elements above the diagonal of L become always positive. Note that dividing u_{ii} in each row to make the diagonal elements of U does not change the sign of each element in the augmented matrix. At the end of backward elimination step, U turns into I and L turns into B^{-1} and all the nonzero elements in B^{-1} are positive.

Below we present a procedure that obtains the location of nonzero elements occurring in B^{-1} without fully inverting the matrix B. The procedure mimics the procedure that transforms the augmented matrix $[B \mid I]$ to $[I \mid B^{-1}]$. However, instead of actually computing numbers, we only specify where nonzero elements occur and instead of treating the augmented matrix, we apply directly on B. At the end, the location of nonzero elements occurring in B^{-1} is marked by '1' in the matrix B.

Procedure 1

1. Given an $n \times n$ diagonally dominant matrix B(we assume that B^T is also diagonally dominant) with $b_{ii} > 0$ for all i and $b_{ij} \leq 0$ for $i \neq j$, replace all the nonzero elements in B by 1.
2. (**Forward process**)
   for $i = 1, 2, \ldots, n-1$
     for $j = i+1, i+2, \ldots, n$
       if $b_{ji} \neq 0$, then add row i to row j and replace nonzero in row j by 1.
3. (**Backward process**)
   for $i = n, n-1, \ldots, 2$
     for $j = i-1, i-2, \ldots, 1$
       if $b_{ji} \neq 0$, then add row i to row j and replace nonzero in row j by 1.

Proof of the Procedure 1. By the properties observed above (2(1), (3)), if b_{ij} is nonzero in the given matrix B, then a positive(nonzero) element also appears at the ijth position of I(i.e., of B^{-1}). Hence, since the value of the nonzero elements is immaterial, one may replace any nonzero element in B by 1 and may work on B directly. This justifies the first item in Procedure 1.

In both forward and backward elimination step, if a zero element b_{ij} becomes nonzero at a certain point, then it remains nonzero until the end of the inversion processes and a positive value appears at the ijth position of B^{-1} because of the properties discussed above (2(1), (3)). Thus, the forward process in Procedure 1 places '1' at the position where a zero element becomes nonzero and keeps the status of nonzero element nonzero. Consequently, the forward process in Procedure 1 gives the location of nonzero elements occurring in L(i.e., below the diagonal of B^{-1}). Similarly, the backward process in Procedure 1 gives the location of nonzero elements occurring below and above the diagonal of B^{-1}.

When the whole process is terminated, 1's in B represents the location of nonzero elements occurring in B^{-1}. □

We illustrate the above procedure using an example below.

EXAMPLE 1. The matrix B given below and B^T are diagonally dominant and $b_{ii} > 0$ for all i.

$$B = \begin{pmatrix} 5 & 0 & 0 & 0 & -1 \\ -1 & 5 & -1 & -1 & 0 \\ 0 & 0 & 5 & 0 & 0 \\ 0 & 0 & -1 & 5 & 0 \\ 0 & -1 & 0 & -1 & 5 \end{pmatrix}. \tag{1}$$

Replacing nonzero element by 1 gives

$$B = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \end{pmatrix}.$$

In forward process, for $i = 1$, $b_{21} \neq 0$. Hence we add row 1 to row 2 and replace nonzero element in row 2 by 1(i.e., we set $b_{25} = 1$). All other elements below b_{11} are zero, so we consider for $i = 2$. Since $b_{52} \neq 0$, we add row 2 to row 5 and replace nonzero in row 5 by 1(i.e., we set $b_{51} = 1$ and $b_{53} = 1$). For $i = 3, 4$, no change occurs. This yields

$$B = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}.$$

In backward process, for $i = 5$, $b_{25} \neq 0$ and $b_{15} \neq 0$. Since all the elements in row 2 in B are already nonzero, we only need to add row 5 to row 1(i.e., we set $b_{12} = b_{13} = b_{14} = 1$). The backward elimination process can be stopped at this point since no operation is needed for $i = 4, 3, 2$. Then the location of nonzero elements(represented by 1's) occurring in B^{-1} is given by

$$B^{-1} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}.$$

If we actually compute the inverse of B given in (1), we would obtain

$$B^{-1} = \begin{pmatrix} 0.2016 & 0.0081 & 0.0035 & 0.0097 & 0.0403 \\ 0.0403 & 0.2016 & 0.0487 & 0.0419 & 0.0081 \\ 0 & 0 & 0.2000 & 0 & 0 \\ 0 & 0 & 0.0400 & 0.2000 & 0 \\ 0.0081 & 0.0403 & 0.0177 & 0.0484 & 0.2016 \end{pmatrix}.$$

The above technique is especially useful in a situation where elements in B^{-1} are extremely small in magnitude such that they are undistinguishable from zero.

3. The expansion of $C(G) = (D - A(G))^{-1}$

Consider an $n \times n$ real matrix $A = (a_{ij})$ with $a_{ij} \geq 0$ for all i and j. If $\sum_{i=1}^{n} a_{ij} < 1$ for all j, then the inverse of $I - A$ can be expressed as an infinite sum of the powers of A:

$$(I - A)^{-1} = I + A + A^2 + A^3 + \cdots . \tag{2}$$

A proof of the convergence of the above expansion is described in Waugh [9].

For a given directed graph $G = (V, E)$ with order n and with the adjacency matrix $A(G) = (a_{ij})$, we let $\sigma = \max_j \sum_{i=1}^{n} a_{ij}$, $j = 1, 2, \ldots, n$. Let $d > \sigma$(or $d \geq n$) and D be the scalar matrix whose diagonal elements are all equal to d, i.e. $D = dI$. Note that choosing such a d or a sufficiently large d makes $B = (D - A(G))$ a strictly diagonally dominant matrix with $b_{ii} > 0$ for all i and $b_{ij} \leq 0$ for $i \neq j$ since $a_{ij} \geq 0$ for all i, j.

Then,

$$(D - A(G))^{-1} = \{d(I - \tilde{A}(G))\}^{-1} = \frac{1}{d}(I - \tilde{A}(G))^{-1},$$

where $\tilde{A}(G) = \frac{1}{d}A(G) \equiv (\tilde{a}_{ij})$ whose elements satisfy $\sum_{i=1}^{n} \tilde{a}_{ij} < 1$ for all j. Hence by (2),

$$\begin{aligned}(D - A(G))^{-1} &= \frac{1}{d}(I + \tilde{A}(G) + \tilde{A}(G)^2 + \cdots + \tilde{A}(G)^k + \cdots) \\ &= \frac{1}{d}(I + \frac{1}{d}A(G) + \frac{1}{d^2}A(G)^2 + \cdots + \frac{1}{d^k}A(G)^k + \cdots).\end{aligned}$$

Letting $C(G) = (D - A(G))^{-1}$, we have

$$\text{(3)} \quad C(G) = \frac{1}{d}(I + \frac{1}{d}A(G) + \frac{1}{d^2}A(G)^2 + \cdots + \frac{1}{d^k}A(G)^k + \cdots).$$

4. Recognition of SCCs

Let $G = (V, E)$ be a directed graph with adjacency matrix $A(G) \equiv (a_{ij})$ and vertex set $V = \{v_1, v_2, \ldots, v_n\}$. We denote $A(G)^k = (a_{ij}^k)$ the kth matrix power of $A(G)$. Then the ijth element a_{ij}^k of $A(G)^k$ represents the number of directed walks of length k from the vertex v_i to v_j. Suppose now that d has been chosen as described in the previous section and $C(G) = (D - A(G))^{-1} \equiv (c_{ij})$ has been computed. Then the following is an immediate consequence of (3).

LEMMA 4.1. *Two vertices v_i and v_j in G are strongly connected if and only if both $c_{ij} \neq 0$ and $c_{ji} \neq 0$.*

Proof. Suppose that $v_i \leftrightarrow v_j$. Then there exists a directed path of length k from v_i to v_j and a directed path of length l from v_j to v_i. This implies that there exists a directed walk of length k from v_i to v_j and a directed walk of length l from v_j to v_i. In other words, $a_{ij}^k \neq 0$ and $a_{ji}^l \neq 0$. From (3), it must be that both $c_{ij} \neq 0$ and $c_{ji} \neq 0$. Conversely, suppose $c_{ij} \neq 0$ and $c_{ji} \neq 0$. From (3), it can be seen that there must exist at least two integers k and l such that $a_{ij}^k \neq 0 \in A(G)^k$ and $a_{ji}^l \neq 0 \in A(G)^l$. This implies that there exists a directed walk of length k from v_i to v_j and a directed walk of length l from v_j to v_i. In other words, there exists a directed path of length $k' \leq k$ from v_i to v_j and a directed path of length $l' \leq l$ from v_j to v_i. Thus, $v_i \leftrightarrow v_j$ in G. □

REMARK 4.2. Lemma 4.1 implies that if either c_{ij} or c_{ji} is zero, then the corresponding vertices v_i and v_j are not strongly connected and hence they can not belong to the same SCC. Thus, whenever $c_{ij} \neq 0$ and $c_{ji} = 0$ happens, one can then set $c_{ij} = 0$.

REMARK 4.3. It can be also perceived that the value of the nonzero elements in $C(G)$ is immaterial; what one needs is the location of the occurrence of nonzero elements of $C(G)$ in order to recognize SCCs.

When $C(G)$ is modified according to the remarks above, then $C(G)$ becomes a symmetric matrix with elements consisting of $1's$ and $0's$. In the rest of this paper, we use the same notation $C(G)$ for the updated one.

The next Lemma shows that the set of vertices corresponding to the column indices of nonzero elements given in each row of $C(G)$ forms a SCC.

LEMMA 4.4. *Let* $T = \{c_{ii}, c_{ij_1}, c_{ij_2}, \ldots, c_{ij_{k-1}}\}$ *be the set of* k *nonzero elements in the* i*th row of* $C(G)$. *Then,* $S = \{v_i, v_{j_1}, v_{j_2}, \ldots, v_{j_{k-1}}\} \subseteq V$ *forms a SCC of* G, *with size* k.

Proof. Let $T = \{c_{ii}, c_{ij_1}, c_{ij_2}, \ldots, c_{ij_{k-1}}\}$. Note that $c_{ii} \neq 0$ for all i. If $k = 1$, then $T = \{c_{ii}\}$ and hence $S = \{v_i\}$ is a trivial point, i.e., a SCC of size 1. So we assume that $\mid T \mid \geq 2$. To show that S is a SCC, it suffices to show that any pair of vertices in S is strongly connected. Let v_p, v_q be an arbitrary pair of vertices in S, where $p, q \in \{i, j_1, j_2, \ldots, j_{k-1}\}$. Two cases occur. The first case is that one of the p and q is i, say $p = i$. Consider c_{ii} and c_{iq} corresponding to the vertices v_i and v_q. Since $c_{iq} \neq 0$, we have $c_{qi} \neq 0$. Thus, by Lemma 4.1 $v_i \leftrightarrow v_q$. The second case is that neither p nor q is equal to i. Consider c_{ip} and c_{iq} corresponding to the vertices v_p and v_q. Since $c_{ip} \neq 0$ and $c_{iq} \neq 0$, we have $c_{pi} \neq 0$ and $c_{qi} \neq 0$. This implies that $v_i \leftrightarrow v_p$ and $v_i \leftrightarrow v_q$. Combining, we have $v_p \leftrightarrow v_q$. Therefore, S is a SCC of G with $\mid S \mid= k$. □

REMARK 4.5. Lemma 4.4 implies that each row of $C(G)$ gives a SCC of G. However, it can be easily observed that row i, row j_1, row j_2, ..., row j_{k-1} all give rise to exactly the same SCC since $v_i, v_{j_1}, v_{j_2}, \ldots, v_{j_{k-1}}$ forms a directed cycle. Hence, once one recognizes a SCC from row i, then one can ignore rows j_1, j_2, ..., j_{k-1} and consider only the remaining rows to recognize another SCC, and so on.

Combining the above results, it can be seen that all of the SCCs of a directed graph G can be recognized by computing the location of the nonzero elements occurring in $C(G)$. A simple procedure for recognizing all of the SCCs of G based on the location of nonzero elements occurring in $C(G) = (D - A(G))^{-1}$ is given below. Note that when using the Procedure 1, we replace nonzero elements in $B = (D - A(G))$ by 1, so we could simply set $a_{ii} = 1$ for all i in $A(G)$ and set $B \leftarrow A(G)$.

Procedure 2

1. Given a directed graph G with its adjacency matrix $A(G)$, set $a_{ii} = 1$ for all i and set $B \leftarrow A(G)$.
2. Use Procedure 1 to compute the location of nonzero elements occurring in B^{-1}, which we denote by $C(G)$.
3. Update $C(G)$ by setting $c_{ij} = 0$ whenever $c_{ji} = 0$ and $c_{ij} \neq 0$, for all $i \neq j$.
4. Recognize all of the SCCs from the column indices of nonzero elements in each row of $C(G)$.

THEOREM 4.6. *The above procedure recognizes all of the SCCs of a directed graph G.*

Proof. By Lemma 4.4, each row of $C(G)$ gives a SCC. Then, considering all the rows of $C(G)$ yields all of the SCCs of G. □

EXAMPLE 2. As an illustration, we consider a directed graph given in Figure 1.

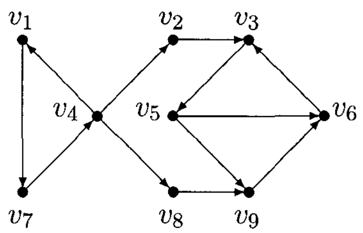

FIGURE 1. A directed graph G with four SCCs

Setting $a_{ii} = 1$ for all i in $A(G)$ and letting $B \leftarrow A(G)$ gives

$$B = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}.$$

Using the Procedure 1, the location of nonzero elements occurring in B^{-1} is given as

$$C(G) = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{pmatrix},$$

and updating $C(G)$ by setting $c_{ij} = 0$ whenever $c_{ji} = 0$ and $c_{ij} \neq 0$ for all $i \neq j$(e.g., since $c_{21} = 0$ and $c_{12} \neq 0$, we set $c_{12} = 0$) yields

$$(4) \qquad C(G) = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{pmatrix}.$$

By considering the column indices of the nonzero elements in row 1 of $C(G)$ given in (4), we obtain $\{v_1, v_4, v_7\}$ as a SCC of size 3. Then, row 4 and row 7 can be ignored since they yield the same SCC. Similarly, we obtain $\{v_2\}$ from row 2, $\{v_3, v_5, v_6, v_9\}$ from row 3(rows 5, 6, and 9 can be ignored), and $\{v_8\}$ from row 8 as the other SCCs of G. That is, $\beta_o(G) = 4$.

5. Conclusions

In this paper we showed that all of the strongly connected components of a directed graph G can be recognized by simply computing the location of nonzero elements occurring in $C(G) = (D - A(G))^{-1}$ without fully inverting $(D - A(G))$. To handle this problem, we developed and presented in the beginning of this paper a simple procedure for computing the location of nonzero elements occurring in B^{-1} without fully inverting B, where B and B^T are diagonally dominant matrices whose diagonal elements are all positive and nondiagonal elements are all nonpositive. The method presented in this paper for recognizing all of the SCCs uses different concept comparing with the previously known methods. It is not linear time method in the order of G. However, the concept of this approach is easy to understand and the approach is simple to describe.

ACKNOWLEDGEMENT. The authors wish to thank the anonymous referee for his/her valuable comments.

References

[1] A. V. Aho, J. E. Hopcroft, and J. D. Ullman, *Data structures and algorithms*, Addison-Wesley, Reading, MA, 1983.

[2] N. Christofides, *Graph theory an algorithmic approach*, Academic Press, London, New York, San Francisco, 1975.

[3] T. H. Cormen, C. E. Leiserson, and R. L. Rivest, *Introduction to algorithms*, MIT Press, Cambridge, MA, 1990.

[4] H. N. Gabow, *Path-based depth-first search for strong and biconnected components*, Inform. Process. Lett. **74** (2000), no. 3-4, 107–114.

[5] G. H. Golub and C. F. Van Loan, *Matrix computations*, The Johns Hopkins University Press, 1983.

[6] Marcu and Danut, *An algorithm for determining the strongly connected components of a finite directed graph*, Stud. Cerc. Mat **28** (1976), no. 1, 57–60.

[7] M. Sharir *A strong-connectivity algorithm and its application in data flow analysis*, Comput. Math. Appl. **7** (1981), no. 1, 67–22.

[8] R. E. Tarjan, *Depth-first search and linear graph algorithms*, SIAM J. Comput. **1** (1972), no. 2, 146–160.

[9] F. V. Waugh, *Inversion of the Leontief matrix by power series*, Econometrica, Vol. **18** (1950), 142–154.

[10] I. Wegener, *A simplified correctness proof for a well-known algorithm computing strongly connected components*, Inform. Process. Lett. **83** (2002), no. 1, 17–19.

DEPARTMENT OF MATHEMATICS, KEIMYUNG UNIVERSITY, DAEGU 704-701, KOREA
E-mail: kmkim@kmu.ac.kr
yykang@kmu.ac.kr