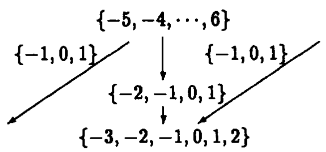
For $\beta = 4$ we can choose:

$$\begin{aligned} r=-2 \quad s=1 \quad &\Longrightarrow \quad D=\{-2,-1,0,1\} \\ &\Longrightarrow \quad D^{*}=\{-2,-1,0,1,2,3,4\} \\ u=-1 \quad v=1 \quad &\Longrightarrow \quad C=\{-1,0,1\} \\ k=-3 \quad l=2 \quad &\Longrightarrow \quad F=\{-3,-2,-1,0,1,2\} \\ &\Longrightarrow \quad E=\{-5,-4,\cdots,6\} \end{aligned}$$

and the conversion in the accumulation of partial products can then be performed as



For a radix-4 systolic multiplier, each cell can then be designed to add a product of two digits from $\{-2,-1,0,1\}$ to a partial sum digit from $\{-3,-2,-1,0,1,2\}$, absorbing and producing a carry from $\{-1,0,1\}$, with resulting sum digit again in $\{-3,-2,-1,0,1,2\}$. □

IV. Conclusion

The literature on computer arithmetic is abundant with examples where digit set conversions are used implicitly in algorithms for performing standard arithmetic operations, but most often without this being realized or at least expressed. This also means that the same problem has been investigated and solved repeatedly. In this paper we have analyzed digit set conversion in its own right, providing, hopefully, a coherent view on conversions into nonredundant as well as into redundant digit sets, and then shown how these fairly general results may be applied in a variety of different arithmetic algorithms. It is our belief that this approach is beneficial because it allows a systematic treatment of a problem that often may be isolated as a subproblem in the context of some algorithm design situation.

Acknowledgment

The author acknowledges constructive comments from the referees and fruitful discussions with D. Matula concerning the proof of Theorem 2.1 and his notation from [6] as used here.

References

[1] M. D. Ercegovac and T. Lang, "On-the-fly conversion of redundant into conventional representations," *IEEE Trans. Comput.*, vol. C-36, no. 7, pp. 895–897, 1987.

[2] R. E. Ladner and M. J. Fisher, "Parallel prefix computation," *J. ACM*, vol. 27, no. 4, pp. 831–838, 1980.

[3] R. P. Brent and H. T. Kung, "A regular layout for parallel adders," *IEEE Trans. Comput.*, vol. C-31, no. 3, pp. 260–264, 1982.

[4] J. Sklansky, "Conditional-sum addition logic," *IRE Trans. Electron. Comput.*, vol. EC-9, pp. 226–231, 1960.

[5] A. Avizienis, "Signed-digit number representations for fast parallel arithmetic," *IRE Trans. Electron. Comput.*, vol. EC-10, pp. 389–400, 1961.

[6] D. W. Matula, "Radix arithmetic: Digital algorithms for computer architecture," in *Applied Computation Theory: Analysis, Design, Modeling*, Raymond T. Yeh, Ed. Englewood Cliffs, NJ: Prentice-Hall, ch. 9, pp. 374–448.

[7] T. Lynch and E. Swartzlander, "A spanning tree carry lookahead adder," *IEEE Trans. Comput.*, vol. C-41, no. 8, pp. 931–939, 1992.

[8] K. Efe, "Multi-operand addition with conditional sum logic," in *Proc. 5th IEEE Symp. Computer Arithmetic*, 1981, pp. 251–255.

[9] T. M. Carter and J. E. Robertson, "The set theory of arithmetic decomposition," *IEEE Trans. Comput.*, vol. 39, pp. 993–1005, 1990.

[10] A. D. Booth, "A signed binary multiplication technique," *Q. J. Mech. Appl. Math.*, vol. 4, pp. 236–240, 1951.

[11] G. W. Reitweiner, "Binary arithmetic," in *Advances in Computers*, vol. 1, NY: Academic Press, 1960, pp. 261–265.

[12] O. L. MacSorley, "High-speed arithmetic in binary computers," *Proc. IRE*, vol. 49, pp. 67–91, 1961.

[13] B. Parhami, "Generalized signed-digit number systems: A unifying framework for redundant number representations," *IEEE Trans. Comput.*, vol. C-39, no. 1, pp. 89–98, 1990.

[14] H. Sam and A. Gupta, "A generalized multibit recoding of two's complement binary numbers and its proof with application in multiplier implementations," *IEEE Trans. Comput.*, vol. 39, pp. 1006–1015, 1990.

[15] S. Vassiliadis, E. M. Schwartz, and D. J. Hanrahan, "A general proof for overlapped multiple-bit scanning multiplications," *IEEE Trans. Comput.*, vol. 38, pp. 172–183, 1989.

Decomposition of $\{0,1\}$-Matrices

R. Swaminathan and D. Veeramani

***Abstract*—A *simple decomposition* of a $r \times c$ $\{0,1\}$-matrix is defined in terms of a collection of disjoint submatrices obtained by deleting a "minimal" set of columns. In general, the number of such simple decompositions is $\Theta(2^r)$. A class of matrices, namely, vertex-tree graphic, is defined, and it is shown that the number of simple decompositions of a vertex-tree graphic matrix is at most $r-1$. Finally, the relevance of simple decomposition to the well-known problem of cluster formation on $\{0,1\}$-matrices is uncovered, and an $O(r^2c)$ time algorithm is given to solve this problem for vertex-tree graphic matrices.**

***Index Terms*—Cluster decomposition, cluster-formation problem, disconnecting set, edge-tree graphic matrix, vertex-tree graphic matrix, simple decomposition.**

I. Introduction

This brief contribution deals with decomposing $\{0,1\}$-matrices into two or more disjoint submatrices by deletion of a "suitable" set of columns. It is shown that the number of ways a $\{0,1\}$-matrix can be decomposed is an exponential function of the number of its rows. Subsequently, a special class of matrices is introduced and it is proved that the number of ways these special matrices can be decomposed is a linear function of the number of its rows. Results obtained on this decomposition are then used to solve the cluster-formation problem (on the previously mentioned special class of $\{0,1\}$-matrices) that deals with the formation of "block-diagonal" structure by permuting rows and columns so that the number of 1's common to two or more of these blocks is minimized.

In this brief contribution, for convenience, trees are equated with their edge sets. In addition, all the matrices are assumed to have at least one 1-entry in each row and column. This is not a

Manuscript received May 21, 1992; revised December 16, 1992.

R. Swaminathan is with the Department of Computer Science, University of Cincinnati, Cincinnati, OH 45221.

D. Veeramani is with the Department of Industrial Engineering, University of Wisconsin, Madison, WI 53706.

IEEE Log Number 9213782.

0018-9340/94$04.00 © 1994 IEEE

$$M = \left[\begin{array}{ccc|c} M_1 & & 0 & \\ \vdots & \ddots & \vdots & D \\ 0 & & M_t & \end{array} \right]$$

Fig. 1. Matrix picture of a decomposition.

restrictive assumption with respect to the decomposition and the cluster-formation problem considered here. Graph theory terminology is, for the most part, consistent with Bondy and Murty [1].

The organization of the brief contribution is as follows. In Section II, the notion of decomposition of $\{0,1\}$-matrices is made precise, and it is shown that the number of ways a given $\{0,1\}$-matrix can be decomposed is an exponential function in the number of its rows. Section III introduces the notion of vertex-tree graphic matrices and shows that the number of ways a vertex-tree graphic matrix can be decomposed is a linear function in the number of its rows. In Section IV, the cluster-formation problem is defined precisely and the relevance of simple decomposition to cluster-formation problem is uncovered. Moreover, an efficient algorithm is described to solve the cluster-formation problem when restricted to vertex-tree graphic matrices. In Section V, the final section, some of the related issues and problems are discussed.

II. Simple Decomposition

In this section, simple decomposition of $\{0,1\}$-matrices is defined precisely, and then it is shown that the number of simple decompositions of a $\{0,1\}$-matrix having r rows is $\Theta(2^r)$. More specifically, the number of simple decompositions is shown to be at most $2^{r-1} - (r+1)$.

Let M be an $r \times c$ $\{0,1\}$-matrix. Associated with M is a bipartite graph $B(M)$ defined as follows. The vertex set of $B(M)$ is $R(M) \cup C(M)$, where $R(M)$ denotes the index set of rows of M and $C(M)$ denotes the index set of columns of M, and $i \in R(M)$ and $j \in C(M)$ are adjacent in $B(M)$ if and only if the ij-entry of M is a 1. The vertex set of a component of B induces a unique submatrix of M, called a *component* of M. The matrix M is *connected* if it has exactly one component. Testing whether M is disconnected, and if so, determining the components of M can be done in $O(r+c+m)$ time (m is the number of 1-entries of M) using an algorithm of Tarjan [2].

Let M be a connected $\{0,1\}$-matrix, and let $B := B(M)$ be its associated bipartite graph. Let D be a proper subset of $C(M)$. If $B \backslash D$ is disconnected (i.e., $B \backslash D$ has at least two components) and no vertex of $R(M)$ is isolated in $B \backslash D$, then D is a *disconnecting set*, abbreviated *d-set*, of M. (Throughout the brief contribution, "\" denotes deletion.) A d-set is *minimal* if it does not properly contain a d-set. Since M is connected, no d-set of M is empty. Moreover, neither a d-set nor a minimal d-set of M is necessarily unique.

Let D be a d-set of a connected $\{0,1\}$-matrix M. Let $R_1, \cdots, R_t$ and $C_1, \cdots, C_t$ be the partitions of $R(M)$ and $C(M) - D$ induced by the components of $B(M) \backslash D$. For $1 \leq i \leq t$, define M_i to be the submatrix of M having row set R_i and column set C_i. If $t \geq 2$, then $\{M_1, \cdots, M_t\}$ is a *decomposition* of M with respect to D. (A matrix picture of a decomposition is given in Fig. 1.) If D is a minimal d-set, then $\{M_1, \cdots, M_t\}$ is a *simple decomposition* of M with respect to D. Observe that each M_i is connected. Moreover, since D is a d-set, by definition, $B(M) \backslash D$ has no isolated vertices. Therefore, each M_i has at least one row and one column.

Not every connected $\{0,1\}$-matrix has a simple decomposition with respect to some minimal d-set. For example, take a $\{0,1\}$-matrix M that has at least two rows and two columns, and add a row of "1"s to M. Then, the resulting matrix does not have a d-set and hence a simple decomposition. In [3], Swaminathan and Wagner showed that if M is a connected $r \times c$ $\{0,1\}$-matrix, then the augmented matrix (M, I) has at least one simple decomposition (with respect to some minimal d-set), where I is a $r \times r$ identity matrix, and that one such simple decomposition of (M, I) can be computed in $O(m)$ time, where m is the number of 1-entries of M. Next, it is shown that there is an infinite class of $\{0,1\}$-matrices, each of which has "exponential" number of minimal d-sets (in the number of rows). In this regard, the following definitions and results are needed.

$$\begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

Fig. 2. A $5 \times 10\{0,1\}$ -matrix having 10 minimal d-sets.

Let $K_n = (U, E)$, for $n > 3$, denote a complete graph on n vertices, and let U be the vertex set and E be the edge set of K_n. Construct a graph G_n from K_n as follows. For each edge $e \in E$, if $e = xy$, then replace e with two new edges $e = xv$ and $f = vy$, where v is a new vertex not in U. Let V denote the set of all such new vertices and F denote the set of all such new edges. It is easy to see that $G_n = (U, V, F)$ is a bipartite graph and is referred to as the *bipartite split* of the complete graph K_n. The vertices of V are referred to as the *split vertices* of G_n.

An *edge cut (vertex cut)* of a connected graph $G = (V, E)$ is a nonempty subset X of $E(V)$ such that $G \backslash X$ is disconnected. A *minimal* edge cut (vertex cut) of G is an edge (vertex) cut no proper subset of which is an edge (vertex) cut.

Theorem 1 below is a folklore-type result; no particular reference is known.

Theorem 1: For $n \geq 2$, the number of minimal edge cuts of K_n is $2^{n-1} - 1$. □

Since for every minimal edge cut of K_n, there is a unique minimal vertex cut W in the bipartite split G_n of K_n such that every vertex of W is a split vertex of G_n, Corollary 2 below follows easily from Theorem 1.

Corollary 2: For $n \geq 2$, the number of minimal vertex cuts of the bipartite split G_n of K_n such that each of these minimal vertex cuts is a subset of the set of split vertices of G_n, is $2^{n-1} - 1$. □

Consider the infinite class of connected $\{0,1\}$-matrices such that for every $\{0,1\}$-matrix M_r in this class, if r denotes the number of rows in M_r, then G_r (bipartite split of K_r) is the associated bipartite graph of M_r. By Corollary 2, G_r has $2^{r-1} - 1$ minimal vertex cuts from its split vertices. Since, exactly r of these $2^{r-1} - 1$ minimal vertex cuts of G_r, when deleted, give isolated vertices, and by definition, every minimal vertex cut J of the associated bipartite graph G_r of M_r such that $G_r \backslash J$ has no isolated vertex, defines a unique minimal d-set of M_r, it follows that the number of minimal d-sets of M_r is $2^{r-1} - (r+1)$. Therefore, given a $r \geq 2$, one can construct a $r \times c$ $\{0,1\}$-matrix, where $c = r(r-1)/2$, having $\Theta(2^r)$ minimal d-sets. (Observe that the number of distinct minimal d-sets in terms of the number of columns c is $\Theta(2^{\sqrt{c}})$, which, also, is quite "large" in number.) Figure 2 shows an example connected $\{0,1\}$-matrix having 5 rows and $2^{5-1} - (5+1)$ minimal d-sets.

Since a connected $\{0,1\}$-matrix, having r rows, can have $\Theta(2^r)$ minimal d-sets, the next natural step is to look for classes of $\{0,1\}$-matrices for which the number of minimal d-sets is bounded by a "low-degree" polynomial in the number of rows. In the following

$$\begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \end{array} \begin{array}{c} \begin{array}{cccccc} a & b & c & d & e & f \end{array} \\ \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{array}$$

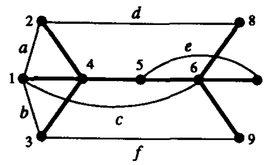
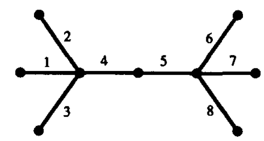


Fig. 3. An example vertex-tree graphic matrix.

$$\begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \end{array} \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

(a)



(b)

Fig. 4. An example edge-tree graphic matrix.

section, such a class of $\{0,1\}$-matrices is introduced.

III. Vertex-Tree Graphic Matrices

In this section, the notion of vertex-tree graphic matrices is introduced, and it is shown that the number of distinct minimal d-sets of a vertex-tree graphic matrix having r rows is at most $r-1$.

A $\{0,1\}$-matrix M is *vertex-tree graphic,* abbreviated *vt-graphic,* if there exists a tree T such that the vertices of T are indexed on the rows of M, and the columns of M are the incidence vectors of the vertex sets of paths of T. (Note that not every path of T need correspond to a column of M.) If such a T exists, then T is a *vertex-tree realization,* abbreviated *vt-realization,* for M. (See Fig. 3 for an example; the edges of vt-realization are shown by thicker lines.) Observe that a vt-realization of a $\{0,1\}$-matrix is not necessarily unique.

Let M be a vt-graphic matrix, and let T be a vt-realization of M. For each column C, add an edge joining the two ends of the corresponding path in T, and call the resulting graph G. Since T is a tree, T and (hence) G are connected. The pair (G,T) is a *graph-tree realization,* abbreviated *gt-realization,* of M. Notice that T is a spanning tree of G and that for each edge e of G not in T, there exists a unique cycle in $T\cup\{e\}$, called a *fundamental cycle* of (G,T). (See Fig. 3 for an example gt-realization; tree edges are shown by thicker lines and the nontree edges are shown by thinner lines.) Observe that a fundamental cycle of (G,T) can also be a loop of G.

A *1-edge cut* of a connected graph G is an edge cut having exactly one edge. The following proposition is easy to prove, and so its proof is omitted.

Proposition 3: Let M be a vt-graphic matrix, and let (G,T) be a gt-realization of M. Then, M is connected if and only if G has no 1-edge cut. □

Proposition 3 implies that if the pair (G,T) is a gt-realization of a connected vt-graphic matrix M, and D is a minimal d-set of M, then $G\backslash D$ has at least one 1-edge cut. Let K be the subset of the edge set of $G\backslash D$ such that $e\in K$ if and only if e is a 1-edge cut of $G\backslash D$. Then, it is easy to see that $K\subset T$. Moreover, if $\{M_1,\cdots,M_t\}$ is the simple decomposition of M owing to the minimal d-set D, then each component of $G\backslash(K\cup D)$ is a gt-realization of a unique M_i for some $1\le i\le t$. The set of edges K is referred to as the 1- *set* of T *associated* with the minimal d-set D. Recall that since D is a d-set of M, $B(M)\backslash D$ has no isolated vertices. Therefore, if $e\in K$ and e is incident to a degree-one vertex u (say) in T, then G has a loop at u.

Lemma 4 below is crucial to the main result of the section.

Lemma 4: Let M be a connected vt-graphic matrix, and let D' and D be two distinct minimal d-sets of M. Let T be a vt-realization of M, and let K' and K, respectively, denote the 1-sets of T associated with D' and D. Then, $K'\cap K=\emptyset$.

Proof: Suppose $K'\cap K\neq\emptyset$. Let $e\in K'\cap K$, and let u and v be the end vertices of e. Let G be a graph such that (G,T) is a gt-realization of M. Since D' is a d-set, for every $d\in D'$, the vertices u and v are in the fundamental cycle $T\cup\{d\}$ in (G,T). Similarly, since D is a d-set, for every $d\in D$, the vertices u and v are in the fundamental cycle $T\cup\{d\}$ in (G,T). This implies that either $D'\subseteq D$ or $D\subseteq D'$. As a result of the minimality of D' and D, it follows that $D'=D$, a contradiction. □

Lemma 4 implies that for every minimal d-set D of M, the associated 1-set is unique. Moreover, the number of disjoint partitions of the edges in T is at most $|T|$ and $|T|\le r-1$. Therefore, Theorem 5 below, the main result of this section, now follows easily from Lemma 4. Notice that the number of minimal d-sets of a vt-graphic matrix does not depend on the number of columns and that a vt-graphic matrix need not have a d-set (and hence a minimal d-set).

Theorem 5: The number of minimal d-sets of a connected vt-graphic matrix, having r rows, is at most $r-1$. □

It is, next, shown that the set of all minimal d-sets of a connected vt-graphic matrix can be constructed in polynomial time. Testing whether a given $r\times c$ connected matrix M is vt-graphic, and if so, constructing a gt-realization can be done in $O(r^2c)$ time using an algorithm in Swaminathan and Wagner [3]. Thus, assume that a gt-realization of M is given. Call it (G,T). For each edge $e=uv$ in T, determine the set of columns D_e in M that have a 1-entry in rows u and v. Testing whether D_e is a d-set of M, and if so, constructing the associated decomposition, can be done in $O(r+c+m)$ using Tarjan's algorithm [2]. (Here, m is the number of 1-entries of M and $m\le rc$.) Thus, all such d-sets corresponding to every edge of T can be constructed in $O(mr)$ time. Finally, among all these d-sets, finding the minimal ones can be done in $O(r^2c)$ time. Therefore, given a connected $r\times c$ $\{0,1\}$-matrix M, testing whether M is vt-graphic, and if so, constructing all the minimal d-sets and their respective simple decompositions can be done in $O(r^2c)$ time.

A $\{0,1\}$-matrix M is *edge-tree graphic,* abbreviated *et-graphic* if there exists a tree T such that the vertices of T are indexed on the rows of M, and the columns of M are the incidence vectors of the edge sets of paths of T. (Note that not every path of T need correspond to a column of M.) If such a T exists, then T is an *edge-tree realization,* abbreviated *et-realization,* for M. (See Fig. 4 for an example.) As in the case of vt-realization, et-realization of a $\{0,1\}$ -matrix is not necessarily unique.

Note that if a connected $\{0,1\}$-matrix is vt-graphic, then it is not necessarily et-graphic, and vice versa, as shown by the following examples. Define a wheel W_n, for $n\ge 3$, to be a graph whose vertex set is $\{v_0,v_1,\cdots,v_n\}$ and edge set is $\{e_1,\cdots e_n,f_1,\cdots f_n\}$ such that $e_i=v_0v_i$ and $f_i=v_iv_{i+1}(\bmod n)$. Choose a spanning tree T_n of G_n to be the edges of W_n incident to the vertex v_0. Let L_n denote the matrix whose rows are indexed on the vertex set of W_n and the columns are the incidence vectors of the vertex sets of paths of T so that for every such path P, there is an unique edge, say e, in $E(W_n)-T_n$ with $P\cup\{e\}$ being a fundamental cycle of (W_n,T_n). Similarly, let M_n denote the matrix whose rows are indexed on the edge set of W_n and the columns are the incidence vectors of the edge sets of paths of T so that for every such path P, there is an unique edge, say e, in $E(W_n)-T_n$ with $P\cup\{e\}$ being a fundamental cycle

$$\begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array}\left[\begin{array}{ccc|ccc} 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & \textcircled{1} \\ \hline 0 & 0 & \textcircled{1} & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{array}\right]$$

Fig. 5. An example cluster decomposition and its exceptional entries.

of (W_n, T_n). Clearly, the matrix L_n is vt-graphic but not et-graphic, and the matrix M_n is et-graphic but not vt-graphic.

Unlike vt-graphic matrices, the class of et-graphic matrices does not have a "small" number of simple decompositions as one may expect. In fact, the example shown in Fig. 2 is an et-graphic matrix which has "exponential" number of distinct simple decompositions in terms of the number of rows. But it is interesting to note that if the given $\{0,1\}$-matrix M is et-graphic, and an et-realization of M is a path, then the number of simple decompositions is at most $r-1$, where r is the number of rows of M. This is true because if M is et-graphic and an et-realization of M is a path, then M is also vt-graphic. In this case, the matrix M is called a *column-consecutive* 1's matrix; analogously, one can also define a row-consecutive 1's matrix. Consecutive 1's matrices were introduced by Fulkerson and Gross [4], and studied further by Booth and Lueker [5]. In particular, using a special data structure called PQ-trees, Booth and Leuker gave a $O(r+c+m)$ algorithm, to test whether a given $r \times c$ $\{0,1\}$-matrix M is a column-consecutive 1's matrix, where m is the number of 1-entries of M.

IV. The Cluster-Formation Problem

In this section, the well-known cluster-formation problem on $\{0,1\}$-matrices is defined precisely, and then a polynomial-time algorithm (in the number of rows and columns) is given to solve the cluster-formation problem for vt-graphic matrices. In the process, the relationship between the cluster-formation problem and simple decomposition is uncovered.

Let M be a connected $\{0,1\}$-matrix, and let $R(M)$ and $C(M)$, respectively, denote its index sets of rows and columns. Assume $t \geq 2$. Let $\{R_1, \cdots, R_t\}$ and $\{C_1, \cdots, C_t\}$, respectively, denote a partition of $R(M)$ and $C(M)$. Then, the submatrix M_i of M whose row index-set is R_i and column index-set is C_i is a *cluster* of M, and the collection $\{M_1, \cdots, M_t\}$ is a *cluster decomposition* of M and is denoted by ϕ. For some $j \in (R(M) - R_i)$ and $k \in C_i$, let m_{jk} denote a 1-entry of M. Then, m_{jk} is an *exceptional entry* and k is an *exceptional column* of M with respect to ϕ. (See Fig. 5 for an example.) The *exceptional count* of a column $k \in C_i$ is the total number of exceptional entries of M with respect to ϕ, and the summation over the exceptional counts of all the columns in C is denoted by N^{ϕ}. For every $1 \leq i \leq t$, if $M_i \in \phi$, after deleting all its exceptional columns owing to ϕ, has at least one column and is connected, then ϕ is a *strict cluster-decomposition* of M.

A $\{0,1\}$-matrix M, for example, can be viewed as part-tool matrix in which the rows are indexed on the set of parts and the columns are indexed on the set of available tools, and an ij -entry of M is a "1" if part i uses tool j; otherwise, it is a "0". Now, cluster decomposability of M (with the associated clusters) describes the similarity among the cutting tool requirements of parts that can be used effectively in determining "good" tool allocations to machines. Therefore, for part-tool matrices, cluster decomposability is a desirable property. See Veeramani [6] for more details.

Now the cluster-formation problem is precisely stated as follows. Let M be a $\{0,1\}$-matrix and Φ (possibly empty) be the set of all strict cluster-decompositions of M. Then, the *cluster-formation problem*, denoted *CFP*, is to find a strict cluster-decomposition that minimizes N^{ϕ} over $\phi \in \Phi$. (Note that if Φ is empty, then CFP is vacuously solved.) There is no algorithm, known in the literature, that solves CFP on $\{0,1\}$ -matrices efficiently, and it is conjectured that in general, CFP is an NP-hard problem. It is shown next that while solving CFP, instead of Φ, it suffices to consider a "special" subset of Φ. Subsequently, such a subset is completely characterized, and for the class of vt-graphic matrices, this special subset of Φ is shown to be computable in polynomial time (in the number of rows and columns). Therefore, CFP, when restricted to the class of vt-graphic matrices, is solvable in polynomial-time.

Recall from Section I that it is easy to check whether a given $r \times c$ $\{0,1\}$-matrix M is connected using the associated bipartite graph of M. Therefore, CFP for disconnected $\{0,1\}$-matrices can be solved in $O(m)$ time, where m is the number of 1-entries of M. Thus, from now on, M is assumed to be connected.

Now, strict cluster-decomposition is related to simple decomposition. Observe that the exceptional columns of the example matrix in Fig. 5 together constitute a minimal d-set, which need not be the case always. Lemma 6 below asserts it suffices to consider only such strict cluster-decompositions while solving CFP.

Lemma 6: Let M be a connected $\{0,1\}$-matrix, and let ϕ^* be a strict cluster-decomposition of M such that $N^{\phi} \leq N^{\phi}$ for all $\phi \in \Phi$. If D^* is the set of exceptional columns of M with respect to ϕ^*, then D^* is a minimal d-set of M.

Proof: Let $\phi^* := \{M_1^*, \cdots, M_t^*\}$ and D_i^* be the set of exceptional columns of M_i^* for $1 \leq i \leq t$. Let $D^* := \{D_1^*, \cdots, D_t^*\}$. Since ϕ^* is a strict cluster-decomposition, it follows that $t \geq 2$ and M_i^* is connected after deleting D_i^* for all $1 \leq i \leq t$. Therefore, D^* is a d-set. If D^* is not minimal, then there exists a d-set D of M such that $D \subset D^*$ and D is minimal. Let $d \in (D^* - D)$. Now partition ϕ^* into two subsets as follows. Since d is an exceptional column and D is a minimal d-set, for at least one cluster M_j^*, for $1 \leq j \leq t$, there is a row $r \in R(M_j^*)$ such that m_{rd} is a 1-entry. Without loss of generality, let $\{M_1^*, \cdots, M_{t'}^*\}$ be all such clusters. Clearly, $t' \leq t$. Since every cluster of ϕ^*, after deleting its exceptional columns, has at least one column and is connected, if $t' = t$, then D is not a d-set of M. Therefore, $t' < t$. Let L^* denote the submatrix of M such that $R(L_i^*) := R(M_1^*) \cup \cdots \cup R(M_{t'}^*)$ and $C(L_i^*) := \{d\} \cup C(M_1^*) \cup \cdots \cup C(M_{t'}^*)$. It is easy to see that $\phi' := \{L^*, M_{t'+1}^*, \cdots, M_t^*\}$ is a strict cluster-decomposition of M such that $N^{\phi'} < N^{\phi^*}$, a contradiction. □

Next, given a simple decomposition of M owing to a minimal d-set D, a strict cluster-decomposition of M whose exceptional count is minimum and the union of whose exceptional columns is equal to D needs to be found efficiently. The following definitions and results given are aimed at achieving this.

Let $\{M_1, \cdots, M_t\}$ be the simple decomposition of M owing to a minimal d-set D. Now, partition the columns of D into $\{D_1, \cdots, D_t\}$ so that $D_i \cap D_j = \emptyset$ for $i \neq j$, and $\cup_{i=1}^{i=t} D_i = D$. (Note that a D_i could be an empty set.) For every $1 \leq i \leq t$, let $M_i' := M_i \cup D_i$. Then, D_i is *assigned* to M_i, and $\{D_1, \cdots, D_t\}$ is an *assignment* of D to the simple decomposition $\{M_1, \cdots, M_t\}$. For a given d-set D, there could be more than one possible assignment of D to $\{M_1, \cdots, M_t\}$, and the set of all assignments of D is called the *assignment set* of D, and is denoted by A_D. Observe that $\{M_1', \cdots, M_t'\}$ is a strict cluster-decomposition of M with D_i being the exceptional columns of M_i'. If N_d^a denotes the exceptional count of $d \in D_i$ owing to the assignment a, then an assignment in A_D that minimizes $\{\sum_{d \in D} N_d^a |$ for all $a \in A_D\}$ needs to be found efficiently. In this regard, the following lemma is needed.

Lemma 7: Let M be a connected $\{0,1\}$-matrix. Let D be a minimal d-set of M and $a^* := \{D_1^*, \cdots, D_t^*\}$ be an assignment of D to the simple decomposition of M with respect to D, which yields $\min\{\sum_{d \in D} N_d^a |$ for all $a \in A_D\}$. Let $a' := \{D_1', \cdots, D_t'\}$ be some assignment of D to the simple decomposition of M with respect to D. Then, for every $d \in D, N_d^{a^*} \leq N_d^{a'}$.

Proof: Suppose for some $d \in D, N_d^{a^*} > N_d^{a'}$. Let $d \in D_i^*$ and $d \in D_j'$ for some $1 \leq i, j \leq t$. Clearly, $i \neq j$. Without loss of generality, assume $i < j$. Let $a'' := \{D_1^*, \cdots, D_i^* - \{d\}, \cdots, D_j^* \cup \{d\}, \cdots, D_t^*\}$. Clearly, a'' is an assignment of D to the simple decomposition of M with respect to D. Moreover, since $N_d^{a^*} > N_d^{a'}$, it follows that $N_d^{a^*} > N_d^{a''}$. Therefore, $\sum_{d\in D} N_d^{a''} < \sum_{d \in D} N_d^{a^*}$, a contradiction. □

Lemma 7 suggests that given a minimal d-set D of M and the cluster decomposition of M with respect to D, finding an assignment $a_D^* := \{D_1^*, \cdots, D_t^*\}$ of D that yields $\min\{\sum_{d\in D} N_d^a |$ for all $a \in A_D\}$ can be done by a simple algorithm as follows. Let $d \in D$ and $\{M_1, \cdots, M_t\}$ be the simple decomposition of M with respect to D. For every $1 \leq i \leq t$, compute the exceptional count of $M_i \cup \{d\}$, and let the minimum exceptional count be due to M_i. Then, add d to D_i. Since $t \leq r$ and $|D| \leq c$, computing a_D^* can be done in $O(rc)$ time, where r is the number of rows and c is the number of columns of M. Therefore, if $\mathcal{D}$ denotes the set of all minimal d-sets of M, then for all $D \in \mathcal{D}$, computing an assignment a_D^* that gives $\min\{\sum_{d\in D} N_d^a |$ for all $a \in A_D\}$ takes $O(rc|\mathcal{D}|)$ time. Once a_D^* has been computed for all $D \in \mathcal{D}$, solving CFP reduces to finding a minimum of $\{\sum_{d\in D} N_d^{a_D^*} | D \in \mathcal{D}\}$, which can be done in $O(|\mathcal{D}|)$ time. Thus, CFP on M can be solved in $O(rc|\mathcal{D}|)$ time.

By Theorem 5, if M is vt-graphic, then $|\mathcal{D}| \leq (r-1)$. Moreover, as shown in Section III, $\mathcal{D}$ can be constructed in $O(r^2c)$ time. Therefore, the following theorem, which is the main result of the section and the brief contribution, follows easily.

Theorem 8: Let M be a $r \times c$ vt-graphic matrix. Then, CFP problem on M can be solved in $O(r^2c)$ time. □

V. Conclusion

It would be interesting to investigate for which other classes of $\{0,1\}$-matrices CFP can be solved efficiently. The *weighted cluster-formation problem* is a generalization of the cluster-formation problem in which each entry of a given $\{0,1\}$-matrix is assigned a weight, and the objective is to find a cluster decomposition that minimizes the summation of weights owing to the exceptional entries. It is straightforward to see that with trivial modifications, the analysis given in this brief contribution for the cluster-formation problem also holds good for the weighted cluster-formation problem. Notice that one could also get another generalization of CFP by removing the condition that every cluster in a strict cluster-decomposition of a $\{0,1\}$-matrix must have at least one column after deleting the set of its exceptional columns. The results of this brief contribution break down without this condition.

Finally, it is not clear whether the results of the brief contribution can be extended to solving a variation of CFP in which the clusters are bounded to have at most certain number of rows and columns. In a manufacturing setting, this translates to the constraint that every cell can locate at most a certain number of machines and handle a certain number of parts. See Kusiak and Chow [7] for details on several variations of the cluster-formation problem.

Acknowledgment

The authors thank Ken Berman and Karthik Mani for their helpful discussions. They also thank the referees for their suggestions in improving the presentation of the brief contribution.

References

[1] J. A. Bondy and U. S. R. Murty. *Graph Theory with Applications.* NY: North-Holland, 1976.

[2] R. E. Tarjan, Depth-first search and linear graph algorithms. *SIAM J. Comput.,* vol. 1, pp. 146–160, 1972.

[3] R. Swaminathan and D. K. Wagner. "On the consecutive-retrieval problem," *SIAM J. Computing,* to appear, 1993.

[4] D. R. Fulkerson and O. A. Gross. "Incidence matrices with the consecutive 1's property." *Bull. Amer. Math. Soc.,* vol. 70, pp. 681–684, 1964.

[5] K. S. Booth and G. S. Lueker. "Testing for consecutive 1's property, interval graphs and graph planarity using PQ-tree algorithms," *J. Compu. Syst. Sci.,* vol. 13, pp. 335–379, 1976.

[6] D. Veeramani, "Physical resource management in large computer controlled manufacturing systems. Ph.D. dissertation, School of Industrial Engineering, Purdue University, West Lafayette, IN, 1991.

[7] A. Kusiak and W. S. Chow. "Decomposition of manufacturing systems," *IEEE Trans. Robotics Automation,* vol. 4, pp. 457–471, 1988.

Identifying Minimal Shift Counters: A Search Technique

Alice M. Tokarnia

***Abstract*— A minimal modulo-m n-stage shift counter is defined as a shift counter with a feedback function of the form $Y_1 = y_n^* \oplus P_1(y_1, y_2, \cdots, y_{n-1}) \oplus \cdots \oplus P_k(y_1, y_2, \cdots, y_{n-1})$, where y_n^* is either y_n or y_n' and $P_i(y_1, y_2, \cdots, y_{n-1})$ is a product of literals of state variables. The feedback function is selected from the set of 2^{2n-1} functions that can be represented in this form so as to minimize the number of product terms, the number of literals of a product term, and the total number of literals, in this order. Due to the shift register properties introduced in this brief contribution, it is possible to identify minimal shift counters using a search technique. Minimal shift counters with up to 14 stages have been identified. Except for very small moduli ($m < 2n$), minimal shift counters can be operated at higher frequencies and require a smaller area than shift counters designed using other methods.**

***Index Terms*—Irreducible shift counters, minimal shift counters, shift counters, shift register properties, synchronous counters.**

I. Introduction

The structure of the n-stage shift registers considered in this brief contribution is shown in Fig. 1. The stages, numbered from 1 to n, are D flip-flops. The *next value* Y_i of the ith stage, for $i \neq 1$, is equal to the *present value* y_{i-1} of the $(i-1)$th stage. The next value Y_i of the first stage is a function $f(y_1, y_2, \cdots, y_n)$ of the present values of the stages. This function has the following form:

$$Y_1 = f(y_1, y_2, \cdots, y_n) = y_n^* \oplus P_1(y_1, y_2, \cdots, y_{n-1}) \oplus \cdots P_k(y_1, y_2, \cdots, y_{n-1}) \qquad (1)$$

Manuscript received August 25, 1992; revised February 24, 1993. This work was supported in part by the Conselho Nacional de Desenvolvimento Cientifico e Technologico (CNPq), Brazil. The results described in this brief contribution are part of the author's Ph.D. dissertation, submitted to the Department of Electrical Engineering, Stanford University, Stanford, CA.

The author is with the Departamento de Engenharia Elétrica, Universidade de Brasília, 70910 Brasilia, DF Brazil.

IEEE Log Number 9213771.