Network Models For Complementary Cell Suppression

Lawrence H. Cox

Complementary cell suppression is a method for protecting data pertaining to individual respondents from statistical disclosure when the data are presented in tabular form. Several mathematical methods for complementary suppression have been proposed in the statistical literature; some have been implemented in large-scale data processing environments by national statistical agencies. Each method has either theoretical or computational limitations. This article presents solutions to the complementary cell suppression problem based on linear optimization over a mathematical network. These methods are shown to be optimal for certain problems and to offer theoretical and practical advantages, including comprehensiveness, comprehensibleness, and computational efficiency.

KEY WORDS: Disclosure limitation; Linear programming; Statistical disclosure; Tabular data.

1. INTRODUCTION

Complementary cell suppression is a standard technique for limiting statistical disclosure in data presented in tabular form, and is the preferred disclosure limitation technique for tables of aggregate magnitude data such as from the U.S. Economic Censuses (Federal Committee on Statistical Methodology 1994). *Cell suppression* involves suppressing from publication the values of all cells representing direct disclosure of confidential data on individual respondents (the *disclosure cells*), together with sufficiently many appropriately selected nondisclosure cells (the *complementary cells*) to ensure that a third party cannot discover confidential respondent data by manipulating linear relationships between released and suppressed table values. The challenge of the cell suppression problem is to select complementary suppressions that provide sufficient disclosure protection while minimizing the amount of information lost due to suppression.

Since the 1970s, the U.S. Census Bureau and other statistical organizations have used large-scale computer systems based on mathematical procedures for complementary cell suppression (Cox, Fagan, Greenberg, and Hemmig 1986; Robertson 1993). These methods involve combinatorial techniques (Cox 1980), linear programming (Robertson 1993), and mathematical networks (Cox 1987, 1993). Each of these methods constructs a complementary cell suppression pattern that protects a single disclosure cell, while attempting to control unnecessary suppression and information loss. The suppression pattern constructed often will protect other disclosure cells partially or even fully. Although the individual mathematical techniques for complementary suppression are different, the methods share some common features. In each case the complementary suppression procedure is performed iteratively—at least once for each cell requiring (further) disclosure protection. Information loss is measured by either number of cells suppressed, total value of suppressed cells, or a combination of these criteria. The iteration is usually implemented in descending order of protection required.

A mathematical network is a specialized linear program defined over a mathematical graph. Networks are widely used for a variety of applications, and standard network optimization software is available. Kelly, Golden, and Assad (1992) offered a network optimization formulation of the cell suppression problem. Formulations based directly on mathematical graph theory have also been investigated (Carvalho, Dellaert, and Osorio 1994; Gusfield 1988). As a rule, graph theoretic methods tend to rely on specialized, often complex, algorithms for which standard software is not available.

None of the mathematical methods for complementary cell suppression solves the problem optimally, due to limitations in either the theoretical model or its implementation. The practical setting for this problem can be extremely large—such as the quinquennial U.S. Economic Censuses—calling for methods that are theoretically compact, flexibly implemented, and computationally efficient. The setting is often organizationally broad—involving many agency personnel with different technical backgrounds—making methods that are easier to understand and maintain desirable.

This article presents an approach to the complementary cell suppression problem—based on mathematical networks—that offers several theoretical and practical advantages. In particular, it solves the complementary cell suppression problem optimally for general tables and for exact disclosure in positive tables under the minimum-number-of-complementary-suppressions criterion; it overcomes the theoretical shortcomings of other minimum-total-value-suppressed methods by basing the optimization on minimum-total-value-suppressed explicitly rather than on a weak surrogate (Sec. 4.3); it enables solutions aimed at protecting entire tables in favor of cell-by-cell protection; and it is based on standard mathematical programming

Lawrence H. Cox is the Senior Mathematical Statistician, U.S. Environmental Protection Agency, National Exposure Research Laboratory (MD-75), Research Triangle Park, NC 27711. The information in this document has been funded wholly or in part by the U.S. Environmental Protection Agency. It has been subjected to Agency review and approved for publication. Mention of trade names or commercial products does not constitute endorsement or recommendation for use. The author acknowledges the helpful comments of a referee.

Journal of the American Statistical Association
December 1995, Vol. 90, No. 432, Theory and Methods

methods that can be implemented, understood, and applied without inordinate institutional investment.

The article is organized as follows. Section 2 delineates the complementary cell suppression problem. Section 3 presents key facts about mathematical networks, and Section 4 presents network models for the principal cell suppression problems. Finally, Section 5 discusses issues related to these methods and methods proposed elsewhere.

2. THE CELL SUPPRESSION PROBLEM

$\mathbf{A}$ denotes a single two-way table, comprising $\mathbf{m}$ internal rows and $\mathbf{n}$ internal columns. $\mathbf{A}$ contains $(m+1)(n+1)$ entries: mn *internal entries,* $\mathbf{a}_{ij}, 1 \leq i \leq m, 1 \leq j \leq n$, arranged as the first m rows and n columns of $\mathbf{A}$; m *row totals,* $\mathbf{a}_{i,n+1}$ arranged as the $(n+1)$st column of $\mathbf{A}$, n *column totals,* $\mathbf{a}_{m+1,j}$ arranged as the $(m+1)$st row of $\mathbf{A}$; and the table *grand total* appearing as entry $\mathbf{a}_{m+1,n+1}$ of $\mathbf{A}$. $\mathbf{A}$ is a *positive table* if $\mathbf{a}_{ij} \geq 0, 1 \leq i \leq m$ and $1 \leq j \leq n$. If this condition is not imposed, then $\mathbf{A}$ is a *general table.* Table 1, a hypothetical positive table with suppressions (from Cox 1993), is used to illustrate the concepts and methods developed herein. The cells marked ($\mathbf{D}$) represent *previously suppressed cells*—disclosure cells and complementary suppressions suppressed at a previous stage in the analysis.

There are many quantitative rules to detect disclosure—i.e., to determine for each cell whether or not the cell should be designated as a disclosure cell (Federal Committee on Statistical Methodology 1994). (See Cox 1981 for the mathematical properties of disclosure rules.) For our purposes, it suffices to assume that a disclosure cell is a cell whose value permits a narrow interval estimate of the contribution to the cell of one or more respondents. Under cell suppression, each disclosure cell (I, J) is suppressed from publication, together with sufficiently many appropriately selected complementary cells to ensure that all interval estimates $V(I, J)$ of a_{IJ} derivable from the published statistical tables are not narrow. To provide a limit on how much to suppress, each disclosure cell is assigned a continuous range of values considered too narrow, called the cell's *disclosure interval.* Disclosure protection for (I, J) is sufficient if all interval estimates of the cell value based on additive relationships between respondent and cell values contain this interval. For our purposes, the disclosure interval for (I, J) will be the symmetric open interval $(a_{IJ} - p_{IJ}, a_{IJ} + p_{IJ})$. $p_{IJ} \geq 0$ is called the *protection limit* for (I, J). It is computed based on the disclosure rule and the respondent values contributing to the cell value (Cox 1981). Because estimates of the values of complementary cells can be used to estimate the values of disclosure cells, each complementary cell is assigned a disclosure interval centered on its value with width equal to its contribution to the protection of (I, J). Modification of the procedures presented here for nonsymmetric disclosure intervals is discussed in Section 5.

Table 1. Hypothetical Table With Suppressions

20(**D**)	10	20	10	20	80
10	10	20(**D**)	5	15	60
40	10	10	20(**D**)	10	90
5	5	15	10(**D**)	5	40
75	35	65	45	50	270

As an example, consider Table 1. Assume that each cell marked "**D**" is a disclosure cell to which a disclosure interval of width equal to 50% of its value has been assigned. (This is much higher than typical protection limits, but will simplify computations.) Then complementary cells must be selected in a manner that ensures that the following inequalities hold in the final table: $10 \leq V(1,1), V(2,3), V(3,4) \leq 30$, and $5 \leq V(4,4) \leq 15$.

This characterizes the most common type of disclosure—*interval disclosure*—under which the value of a disclosure cell is protected to within a continuous interval. The other type of disclosure is *exact disclosure,* for which only the precise value of the cell is protected. Exact disclosure is typically used for frequency count data (which are integers), and in that case $p_{IJ} = 1$. Interval disclosure is used in the U.S. Economic Censuses, with the exception of the Census of Agriculture, which uses exact disclosure (Cox 1993).

Most mathematically based cell suppression methods are *single-cell* methods, meaning that they provide sufficient disclosure protection to a single suppressed cell at a time and protect the entire statistical table or set of tables by applying the single-cell method iteratively to each suppressed cell that remains unprotected. The main requirement of a single-cell suppression method is that it provide sufficient disclosure protection for the target cell. A second requirement is that it do so with minimum information loss due to the suppression. Information loss can be measured in many ways, the most typical being the number of additional cells suppressed as complementary suppressions to protect the subject cell and the total value of additional complementary suppressions. These are the *minimum-number-of-complementary-suppressions* criterion and the *minimum-total-value-suppressed* criterion.

The development of complementary cell suppression methods is motivated by the practical considerations of statistical confidentiality within data collection organizations. Under these conditions, all problems considered here admit at least one feasible solution—suppress all internal and totals entries in the table. This solution is of course undesirable, but its existence assures that the procedures presented here will converge to a solution.

3. PROPERTIES OF MATHEMATICAL NETWORKS

A mathematical network is a type of linear program that is uniquely suited to problems involving two-way statistical tables. Only essential properties of networks are presented here. The reader is referred to the books of Glover, Klingman, and Phillips (1992) and Kennington and Helgason (1980) and to other standard texts on network optimization for further information.

A *mathematical network* consists of a set of objects called *nodes* together with a set of objects called *arcs* defined between ordered pairs of nodes. Nodes are denoted by letters such as P and Q and represented graphically as points. Arcs are denoted by ordered pairs of distinct nodes, (P, Q) and

(Q, P), and represented graphically as arrows pointing from the first node to the second node of the pair. An arbitrary but consistent notion of direction between nodes is established and arcs oriented in that direction are called *positive,* whereas arcs in the opposite direction are called *negative;* the corresponding nodes are referred to as the *from-node* and the *to-node* of the directed arc. A network on a set of nodes can involve none, some, or all of the possible positive and negative arcs that can be created between the nodes.

Networks represent linear relationships of a specialized form: *aggregation* relations. Assignments or movements of numeric quantities within a network are referred to as *network flows.* Flows proceed between network nodes along network arcs in the direction of the arc. Each node is assigned a value known as the *node requirement.* A positive requirement indicates the amount net outflow required at the node (sum of flows along arcs directed out of the node minus sum of flows directed into the node), a negative requirement indicates net inflow required at the node, and a zero node requirement indicates balanced inflow and outflow. Row and column nodes are denoted by i and j (or, I and J to particularize a node), and arc flows are denoted by $\mathbf{x}_{ij+}$ and $\mathbf{x}_{ij-}$, relative to the direction of flow. Networks to solve various forms of the complementary cell suppression problem are constructed in Section 4.

The connection between networks and mathematical programs is as follows. $\mathbf{B}$ denotes the *node-arc incidence matrix* of the network: $\mathbf{B}$ contains one row for each node and one column for each arc. The arc-column entry corresponding to the from node of the arc equals $+1$; the entry corresponding to the to node equals -1; and all other entries in the arc-column equal zero. Here $\mathbf{x}$ denotes the column vector of variables corresponding to the arc flows, and $\mathbf{R}$ denotes the row vector of node requirements. The linear constraint system represented by the network is $\mathbf{Bx} = \mathbf{R}, \mathbf{x} \geq \mathbf{0}$. In some applications the $\mathbf{x}$ values are further restricted by upper *capacity constraints* of the form $\mathbf{x} \leq \mathbf{u}$, where $\mathbf{u}$ is a column vector of nonnegative upper limits on flows along individual arcs. Uncapacitated flows have $u = \infty$.

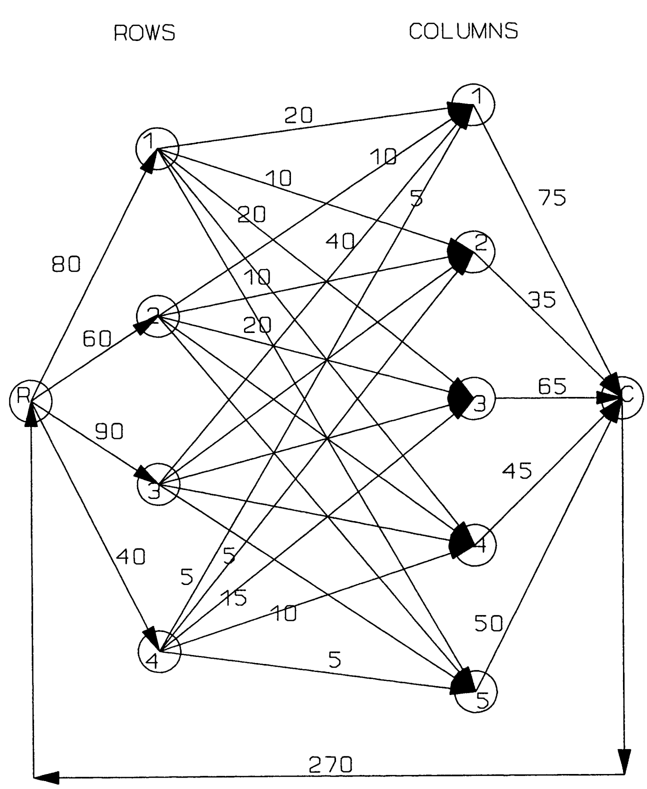

Figure 1. Network for Table 1. Opposing arcs are not drawn.

The close relationship between networks and two-way tables is illustrated in Figure 1, which represents Table 1 (without its suppression information) as a network. Oppositely directed arcs are not drawn.

A *network optimization problem,* $(N, \mathbf{c})$, consists of a network N together with a cost function to be minimized subject to the linear constraints imposed by the network structure; $\mathbf{c}$ denotes the row vector of arc costs. The mathematical statement of the (upper) *capacitated network optimization problem* is

$$\min \mathbf{cx} \quad \text{subject to} \quad \mathbf{Bx} = \mathbf{R}, \qquad \mathbf{0} \leq \mathbf{x} \leq \mathbf{u}. \quad (1)$$

Assume that N is the network representation of a positive table $\mathbf{A}$. Given an arc of N [for definiteness, arc (Row I, Col J, $+$)], a *circuit* γ in N containing this arc is a sequence of distinct arcs $\{\alpha_1, \alpha_2, \ldots, \alpha_k\}$ of N of length k satisfying: $\alpha_1 = (I, J, +)$; the to-node of each arc equals the from-node of the successive arc; and the to-node of α_k equals the row I node (viz., α_1 is treated as the successor of α_k). If successive arcs in γ are in opposite directions (hence, k is even), then γ is an *alternating cycle.* Define $g(\gamma) = \min\{a_{ij}$: $(i, j, +$ or $-) \in \gamma\}$. A flow of up to $g(\gamma)$ units is possible along γ without violating the constraint $\mathbf{x} \geq 0$, viz., add $g(\gamma)$ to each positive arc on γ and subtract $g(\gamma)$ from each negative arc. By reversing arc directions, flow up to $g(\gamma)$ units in the reverse direction is possible. Then $V(I, J)$ can assume any value in the interval $[a_{IJ} - g(\gamma), a_{IJ} + g(\gamma)]$ (similarly for any other cell involved in γ). For example, an alternating cycle for Table 1 involving cell $(1, 1)$ is (Row 1, Col 1, $+$), (Row 2, Col 1, $-$), (Row 2, Col 3, $+$), (Row 3, Col 3, $-$), (Row 3, Col 4, $+$), (Row 1, Col 4, $-$); flow of up to $g(\gamma) = 10$ units can be made along this alternating cycle in either direction. This example illustrates the following key connection between alternating cycles in a network and complementary suppression in a table.

Observation. A disclosure cell (I, J) in a table with protection limit p_{IJ} is protected if there exists an alternating cycle γ comprising only suppressed cells that contains (I, J) and satisfies $g(\gamma) \geq p_{IJ}$.

Networks enjoy a property that is crucial to the formulations of the next section.

Integrality Property. If $\mathbf{R}$ and $\mathbf{u}$ are integer valued and $\mathbf{cx}$ is nonconstant, then any optimal solution $\mathbf{x}$ is also integer valued.

The integrality property will be used powerfully: If $\mathbf{u} \leq 1$ is integer valued, then both $x_{ij+}, x_{ij-} = 0$ or 1 for all (i, j). This formulation enables the use of continuous, computationally efficient network optimization methods to solve a dichotomous decision problem—the complementary cell suppression problem.

For the constructions in Section 4, it is often not necessary to distinguish between positive and negative arcs. In such cases, the notation $\mathbf{x}_{ij*}$ is used to denote either $\mathbf{x}_{ij+}$ or $\mathbf{x}_{ij-}$, or both. Analogous notation is used for other quantities.

4. A DICHOTOMOUS NETWORK MODEL FOR CELL SUPPRESSION

4.1 Network for Single-Cell Suppression in a Single Table

The cell suppression problem for a single table $\mathbf{A}$ subject to a standard information loss criterion such as minimum-number-of-complementary-suppressions or minimum-total-value-suppressed can be modeled as a dichotomous network. As single-cell suppression methods protect one cell at a time, the network is optimized once for each cell requiring disclosure protection. S denotes the set of previously suppressed cells, and $\#S$ denotes the number of cells in S. $(I, J) \in S$ denotes the *target cell*—the suppressed cell being disclosure protected at the current iteration. $U \subseteq S$ denotes the set of table cells that remain unprotected; $\bar{S} = S \sim (I, J)$.

The network N described here is solved once for each unprotected suppressed cell $(I, J) \in U$. At each iteration, (I, J) and perhaps additional unprotected suppressed cells $(I^*, J^*) \in U$ are protected through the creation of an alternating cycle containing (I, J). The network N for Table 1 and target cell $(1, 1)$ is illustrated in Figure 2.

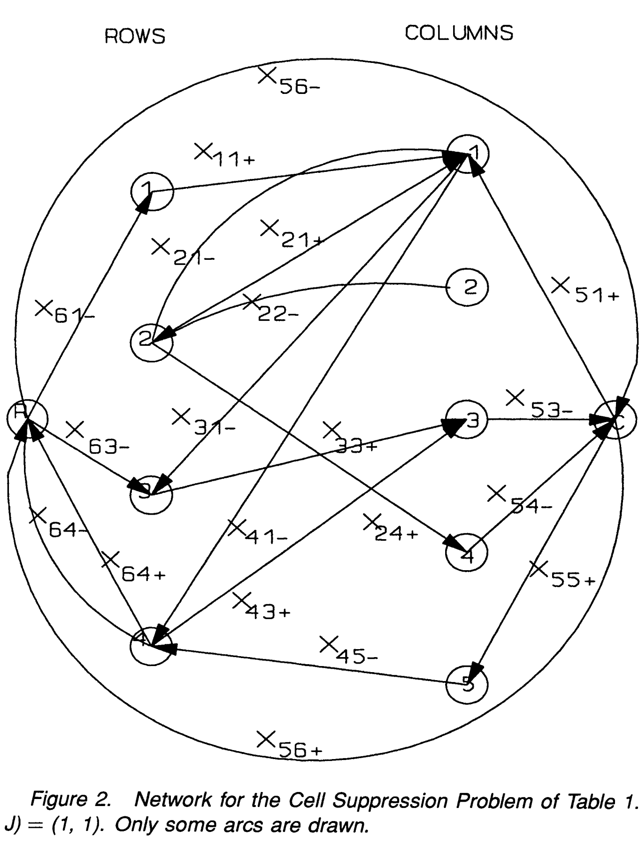

Figure 2. Network for the Cell Suppression Problem of Table 1. (I, J) = (1, 1). Only some arcs are drawn.

Nodes. There are $m + n + 2$ nodes. The first m nodes correspond to the m internal rows of $\mathbf{A}$, the next n nodes correspond to the n internal columns of $\mathbf{A}$, and the last two nodes correspond to the column of row totals adding to the grand total $\mathbf{a}_{m+1,n+1}$ of $\mathbf{A}$ and the row of column totals adding to $\mathbf{a}_{m+1,n+1}$.

Arcs. There are $(m+1)(n+1)$ positive arcs: one from each row node to each column node, one from each row node to the first grand total node, one from the second grand total node to each column node, and one from the second grand total node to the first grand total node. Similarly, between the same node pairs are $(m+1)(n+1)$ oppositely directed negative arcs. The corresponding flows are denoted by $\mathbf{x}_{ij+}$ and $\mathbf{x}_{ij-}, 1 \le i \le m+1, 1 \le j \le n+1$.

Node requirements. All node requirements equal zero.

Arc capacities. The capacities of x_{IJ+} and x_{IJ-} corresponding to the target suppressed cell (I, J) are $u_{IJ+} = 1$ and $u_{IJ-} = 0$. If $a_{ij} = 0$, then $u_{ij*} = 0$. Unless explicitly stated otherwise, $u_{ij*} = 1$ otherwise.

Zero cells often are recognizable even if suppressed and are rendered ineligible for suppression by their corresponding zero arc capacities. If suppressed zero cells are not recognizable, then exceptions may be made to this rule (e.g., in general tables). Another exception, for positive tables, is when upper and lower protection are provided separately: alternating cycles may pass through a zero cell (i, j) in the positive direction, so that only $u_{ij-} = 0$ is required during the upper protection phase. In practice, zero capacity arcs are removed from the network. A relaxation of imposing zero capacities on zero cell arcs is to assign them high cost.

Most software for network optimization facilitates the assignment of nonzero lower capacities for arcs, usually by creating additional arcs. Lower capacities can be used to specify nonzero flow along certain arcs (e.g., by setting lower and upper arc capacities as equal). For simplicity, lower capacities are avoided here; and the cost function is used for these purposes.

It remains to specify a cost function for the network optimization problem (N, c). Cost functions vary with the specifics of the cell suppression problem being considered. But unless stated otherwise, all problems share the following costs.

Arc costs. The cost along arc x_{IJ+} corresponding to the target suppressed cell is $c_{IJ+} = -(c_0 + 1)$, where $c_0 = \sum_{(i,j)\neq(I,J)}(c_{ij+} + c_{ij-})$; $c_{IJ-} = 1$. For $(I^*, J^*) \in \bar{S}, c_{I^*J^*+} = c_{I^*J^*-} = 1$. For $(i, j) \notin S$, arc costs depend on the problem type but are subject to $c_{ij+}, c_{ij-} \ge \#S$.

Total cost. The total cost, $\mathbf{c}(\mathbf{x}) = \sum_{i,j}(c_{ij+}x_{ij+} + c_{ij-}x_{ij-})$, is minimized by the network optimization.

The integrality property of networks and the $\{0, 1\}$ capacities ensure that each arc flow in the optimal solution satisfies $x_{ij*} = 0$ or 1. This permits the network optimization and complementary cell suppression problems to be connected by the following rule.

Cell suppression rule. Suppress cell (i, j) if $x_{ij+} = 1$ or $x_{ij-} = 1$ in the optimal solution.

Recall that all flows x_{ij*} are nonnegative. The large negative arc cost c_{IJ+} forces $x_{IJ+} = 1$, avoiding the trivial solution $\mathbf{x} = \mathbf{0}$ and fulfilling the requirement to suppress the target cell (I, J). The zero node requirements impose the constraints

$$\sum_{j=1}^{n} x_{ij+} = \sum_{j=1}^{n} x_{ij-}, \qquad 1 \le i \le m+1$$

and

$$\sum_{i=1}^{m} x_{ij+} = \sum_{i=1}^{m} x_{ij-}, \qquad 1 \le j \le n+1. \tag{2}$$

The conditions (2) and $x_{IJ+} = 1$ ensure that an optimal solution to (N, c) is an alternating cycle γ containing (I, J). The arc capacity $u_{IJ-} = 0$ forces $x_{IJ-} = 0$, ensuring that the cycle is nontrivial (i.e., $k \ge 4$). This cycle is called the *protection cycle* for the target cell (I, J).

The only negative arc cost is c_{IJ+}. The next smallest costs are $c_{I*J*+} = c_{I*J*-} = 1$ for previously suppressed cells $(I^*, J^*) \in \bar{S}$. Together with $c_{ij+}, c_{ij-} \ge \#S$ for $(i, j) \notin S$, this encourages the network optimization to select previously suppressed cells as complementary suppressions for the target cell (I, J). As $c_{ij+}, c_{ij-} > 0$ for $(i, j) \ne (I, J)$, trivial subcycles (i.e., $x_{ij+} = x_{ij-} = 1$) are avoided, ensuring no superfluous suppressions in the optimal solution.

4.2 Optimizing Single-Cell Suppression Under the Minimum-Number-of-Complementary-Suppressions Criterion

4.2.1 General tables. For general tables, $p_{IJ} < \varepsilon$ for any positive quantity ε. Define $g(\gamma) = \infty$.

Arc costs: general table. $c_{ij+} = c_{ij-} = \#S$ for all $(i, j) \notin S$.

Thus, the total cost $c(x)$ in an optimal solution $\{\mathbf{x}^*\}$ for a general table under the minimum-number-of-suppressions criterion satisfies

$$c(x^*) = (\#S)r + s - (c_0 + 1), \tag{3}$$

where $\mathbf{r}$ equals the minimum number of complementary suppressions needed to protect the target cell (I, J) and s equals the number of previously suppressed cells appearing in the optimal protection cycle γ. This solution is guaranteed by how the arc costs were *stratified.* The large negative cost forces the suppression of (I, J); the large positive cost ensures that r is minimized; and, the unit costs ensure that any previously-suppressed cell that can be used (while minimizing r) will be used, but only once (viz., it will not create a trivial subcycle).

The optimal solution contains s previously suppressed cells $(I^*, J^*) \in \bar{S}$ and r new complementary cells $(i, j) \notin S$. If $s > 0$, then each of the s cells $(I^*, J^*) \in \bar{S}$ contained in the protection cycle γ for (I, J) is also protected by γ and does not require a separate network optimization. If $r = 0$, then γ involves only previously suppressed cells, and the network has verified that target cell (I, J) is protected.

The network optimization (N, c) solves the cell suppression problem for general tables optimally under the minimum-number-of-suppressions criterion, as it produces a solution involving precisely r complementary suppressions.

There typically are many optimal solutions under the minimum-number-of-suppressions criterion, and it is desirable to select from among these solutions one that suppresses the least total amount of information (Cox 1980), as measured for each cell by some function f satisfying $0 \le f_{ij} \le 1$ (e.g., $f_{ij} = (|a_{ij}| + 1)/(a + 1)$ for $a = \sum_{i,j} |a_{ij}|$). This can be done using the same network N with a new cost function $c'(\mathbf{x})$, with costs

$$c'_{ij*} = (F + 1)c_{ij*} + f_{ij}, \quad \text{for} \quad F = \sum_{i,j} f_{ij}. \tag{4}$$

The network optimization (N, c') solves the complementary cell suppression problem for general tables optimally with respect to this *two-stage information loss criterion.* In addition, if the iteration of network optimizations is done in the order of decreasing protection required, then each $(i, j) \ne (I, J)$ selected as a complementary suppression at one iteration satisfies $p_{ij} \le p_{IJ} \le g(\gamma)$, and thus has received sufficient disclosure protection from γ and need not be reexamined by means of a separate iteration.

4.2.2 Exact Disclosure in Positive Tables. The network optimization (N, c) of Section 4.2.1 optimally solves the complementary cell suppression problem for exact disclosure in positive tables under the minimum-number-of-complementary-suppressions criterion. (N, c') optimally solves this problem under the two-stage criterion. Moreover, the protection cycle for target cell (I, J) protects all complementary suppressions made at the current iteration, thus avoiding unnecessary iterations.

4.2.3 Interval Disclosure in Positive Tables. The network optimization (N, c') incorporates the cell value a_{ij} into the cost function as a refinement of (N, c) for the case of exact disclosure in positive tables. A similar but more refined approach is needed for the case of interval disclosure.

Optimal methods are not available for interval disclosure in positive tables, nor for the minimum-total-value-suppressed criterion in general and positive tables. This stems from two factors. The first is the difficulty of incorporating both dichotomous decision variables and continuous variables representing protection levels into a single, computationally efficient linear programming formulation. The second is that the complementary cell suppression problem under the minimum-total-value-suppressed criterion is *NP hard,* suggesting that the existence of an efficient (polynomial time) algorithm for this problem is unlikely (Kelly, Golden, and Assad 1992). The solutions presented here and by others for these problems use a single-cell suppression method to attempt to create a cycle containing the target cell (I, J) that provides sufficient disclosure protection for (I, J). If such a cycle does not exist or is not found, then a partial solution is constructed, and the method is repeated until sufficient protection has been provided for the target cell. Because each iteration is based on previous partial constructions, these iterative methods are heuristic and cannot be guaranteed to produce an optimal solution—either for the target cell or for the entire table. Iterative methods for this problem are also vulnerable to worst-case behavior in well-contrived examples. However, a good iterative method will perform well on average and can be modified to accommodate worst cases.

The following methods presented and those in Section 4.3 are heuristic iterative methods. The performance of each method will depend on the characteristics of the tables and disclosure situations to which it is applied. The methods presented here are intended to illustrate the application of mathematical networks to complementary cell suppression. In a specific application, best results may be obtained by using one method exclusively, by specializing it, or by using several methods in tandem. For example, for a small number of tables of great importance, several methods might be applied to each table and the best results selected for each table individually. In a large census, competing methods might be evaluated using a representative set of test tables, resulting in the selection of a single method to be applied to all released tables.

The preceding optimal solutions provide sufficient disclosure protection by means of a single cycle of nonzero cells containing the target cell. Interval disclosure in positive tables also requires that the cycle(s) through target cell (I, J) permit a total flow of at least p_{IJ} units in each direction. Depending on the relative sizes of the a_{ij}, this may require more than one cycle through (I, J). The iterative step in our heuristic procedure attempts to provide sufficient protection along a single cycle whenever possible, as follows.

Arc capacities and costs. Modify N to create N' satisfying $u_{ij*} = 0$ whenever $u_{ij*} = 1$ in N and $a_{ij} < p_{IJ}$. For $(i, j) \in \bar{S}, c_{ij*} = 1$; for $(i, j) \notin S, c_{ij*} = \#S$.

Thus only cells that permit the required flow in both directions are considered as potential complementary suppressions. The optimization (N', c) will construct a single sufficient protection-cycle for target cell (I, J) whenever such exists. This cycle will be optimal in that it contains the minimum number of complementary suppressions possible among all sufficient single-cycle solutions. There could exist a combination of protection cycles that is sufficient and involves fewer complementary suppressions, but in general this is unlikely.

Because it is easy to contrive examples exhibiting worst-case behavior for each information loss criterion, a two-stage optimization that minimizes total value suppressed as a secondary criterion is often desirable. Using (4), this is obtained by the replacement,

$$c_{ij*} \to (1+F)c_{ij*} + a_{ij}, \quad \text{where} \quad F = \sum_{i,j} a_{ij}. \quad (5)$$

An heuristic iterative procedure, $\mathbf{IP(I, J)}$, for achieving sufficient complementary disclosure protection for (I, J) under interval disclosure in positive tables is as follows:

Step 1. Compute an optimal solution $(\mathbf{x}, \gamma)$ to (N', c).
Step 1a. If no solution exists, compute an optimal solution $(\mathbf{x}, \gamma)$ to (N, c).
Step 2. Compute $g(\gamma) = \min\{a_{ij}\colon (i, j) \in \gamma\}$.
Step 3. Adjoin to S all cells $(i, j) \in \gamma, (i, j) \neq (I, J)$; set $p_{ij} = \min\{p_{IJ}, g(\gamma)\}$.
Step 4. If $g(\gamma) \geq p_{IJ}$, then END.
Step 5. Replace p_{IJ} by $p_{IJ} - g(\gamma)$.
Step 6. For $(i, j) \in \gamma, (i, j) \neq (I, J)$, set $u_{ij*} = 0$.
Step 7. GOTO Step 1.

If an optimal solution is found in Step 1, then the current iterative stage comprises one network optimization on N'. If no solution is found, then Step 1a performs a second optimization on the larger network N. These two optimizations can be combined using cost stratification into one optimization, as follows. Define $L_1 = \#\{(i, j) \in \bar{S}\colon a_{ij} \geq p_{IJ}\}$ and $L_2 = \#\{(i, j) \notin S\colon 0 < a_{ij} < p_{IJ}\}$. Define costs $\{c'\}$ satisfying

$$\begin{aligned} c'_{ij*} &= -\left(1 + \sum_{(i,j)\neq(I,J)} c'_{ij*}\right) \qquad \text{if } (i,j) = (I,J), \\ &= 1 \qquad \text{if } a_{ij} \geq p_{IJ} \text{ and } (i,j) \in \bar{S}, \\ &= \#S \qquad \text{if } a_{ij} \geq p_{IJ} \text{ and } (i,j) \notin S, \\ &= \#S(1+L_1) \qquad \text{if } a_{ij} < p_{IJ} \text{ and } (i,j) \in \bar{S}, \\ &= \#S(1+L_1+L_1L_2+L_2) \qquad \text{if } a_{ij} < p_{IJ} \\ &\quad \text{and } (i,j) \notin S. \end{aligned} \quad (6)$$

Replace Steps 1, 1a and 7 in iterative procedure $IP(I, J)$ by

Step 1′. Compute an optimal solution $(\mathbf{x}, \gamma)$ to (N, c').
Step 7′. GOTO STEP 1′.

The two procedures will not necessarily produce the same results. The combined procedure is more efficient but also more complex, and it is likely to produce large costs. Numerical overflow can be avoided by replacing arc costs c_{ij*} based on a_{ij} by smaller but still monotonic stratified costs based on values such as $\log(1 + a_{ij})$. Two-stage optimization (i.e., minimizing total-value-suppressed among minimum-number-of-complementary-suppressions solutions) can be achieved using costs analogous to (5).

4.3 Optimizing Single-Cell Suppression Under the Minimum-Total-Value-Suppressed Criterion

A sufficient complementary cell suppression pattern for suppressed cell (I, J) that minimizes total value suppressed is obtained from the network optimization (N, c), where

$$\begin{aligned} c_{ij*} &= -\left(1 + \sum_{(i,j)\neq(I,J)} c_{ij*}\right) \qquad (i,j) = (I,J), \\ &= 1 \qquad (i,j) \in \bar{S}, \\ &= \#S + a_{ij} \qquad (i,j) \notin S. \end{aligned} \quad (7)$$

Other heuristic solutions to the single-cell minimum-total-value-suppressed complementary suppression problem have been reported (Kelly, Golden, and Assad 1992). Large-scale implementation of heuristic procedures based on linear optimization have been implemented for economic and agriculture censuses and surveys by the U.S. Census Bureau and Statistics Canada (see Cox 1993 and Robertson 1993, for details). The Census Bureau method is based on network optimization in which the flow variables, z_{ij*}, represent the disclosure protection achieved by suppressing cell (i, j); arc costs, d_{ij*}, are based on the cell value a_{ij}. The cell is suppressed if either $z_{ij+}, z_{ij-} > 0$ in the optimal solution. The Statistics Canada method is based on general linear programming and uses the same flow variables, z_{ij*}, and decision rule, but arc costs d_{ij*} are based on a logarithmic function of a_{ij}. Both methods seek minimum-total-value-suppressed solutions. But in each case the cost function $d(z) = \sum_{i,j} d_{ij*} z_{ij*}$ is not a good surrogate for minimum-total-value-suppressed $c(x) = \sum_{i,j} a_{ij*} x_{ij*}$; whereas, $c(x)$ appears explicitly in the cost formulations here. This is a theoretical improvement.

$c(x)$ can be used to seek optimal solutions. If all $a_{ij} \geq p_{IJ}$ for all nonzero cells, then the optimization (N, c) provides an optimal solution to the minimum-total-value-suppressed problem. In the presence of many cell values $a_{ij} < p_{IJ}, (N, c)$ is likely to create suppression patterns that provide insufficient protection, requiring iteration leading to suppression of additional information. A good approach is to use $c(x)$ in conjunction with the iterative procedure IP(I, J) (viz., seek first solutions involving only cells satisfying $a_{ij} \geq p_{IJ}$, form partial solutions as necessary, and then iterate the method). Alternatively, the double optimizations Steps 1 and 1a can be combined using a cost function that places cells with $a_{ij} < p_{IJ}$ in a higher cost stratum, as in (6).

4.4 Network Model for Cell Suppression in Tables Related Hierarchically

Cox and George (1989) presented a method for representing a two-way table subject to subtotals constraints along its rows (or columns) as a mathematical network. The construction, for column subtotals, is as follows. Begin at the first grand total node and disaggregate the grand total by negative arcs connected to each row node. Disaggregate each row total by positive arcs to each column node. These arcs correspond to the internal cells of the table. Connect the column nodes by negative arcs to nodes representing the lowest level column subtotals. Connect these nodes by positive arcs to nodes representing the second–lowest-level column subtotals, and so on, reversing the direction of new sets of arcs at each stage, until nodes representing the column totals of the full table are reached. Connect these nodes to the second grand total node. If the number of levels of subtotals is even, then connect the two grand totals nodes by a positive arc from the second to the first grand total node. If it is odd, then to maintain consistency in assigning arc directions, create a dummy node between the two grand totals, and connect this node from the second grand total node by a negative arc and to the first grand total node by a positive arc. At the conclusion of the construction, an identical set of oppositely directed arcs is created at every node except the dummy node. Denote this network as H.

The primary advantages of combining hierarchically related tables within a single cell suppression network are to increase the pool of candidate complementary suppressions for each disclosure cell and thereby improve the computational efficiency of the entire disclosure limitation process. Methods equivalent to the Cox and George (1989) formulation were implemented in the 1992 U.S. Economic Censuses to incorporate hierarchical Standard Industry Code (SIC) classifications within a single network (Cox 1993). For the Economic Censuses, tables defined at the same single level of geographic structure (e.g., aggregate data at the county level adding to state-level totals) were combined hierarchically along the SIC dimension; a network was constructed beginning with nodes representing county-level aggregates across all industries to nodes representing county-level aggregates for industries within individual two-digit SIC groups; from two-digit SIC nodes to three-digit SIC nodes; and so on, until the lowest level of SIC aggregation (in general, six-digit SIC) was reached. These nodes were then reconnected to the beginning nodes through state-level nodes as described previously. Recent progress has been reported on systematic extension of these methods to non-hierarchical aggregation structures (D. Robertson, personal communication, August 23, 1994).

The methods of Sections 4.1–4.3, presented for a single two-way table, apply equally to the network H corresponding to a set of hierarchically related two-way tables. For single tables, and particularly for hierarchically related tables, it is usually desirable to minimize the suppression of (higher-level) totals relative to that of lower-level entries. This can be accomplished by creating higher cost strata for arcs corresponding to cells at higher aggregation levels.

4.5 Multiple-Cell Complementary Suppression

Single-cell complementary suppression procedures require at least one optimization for each cell (I, J) requiring disclosure protection. The suppression algorithm is applied iteratively until a complete pattern—equal to the union of patterns constructed at each iterative stage—is achieved. After each stage, or after the last stage, the pattern is examined to release superfluous complementary suppressions. The final pattern equals the complete pattern minus superfluous suppressions. Because this is an iterative process, the final pattern may not be optimal (viz., a lower-cost, complete pattern may exist). The question arises: Does there exist an (optimal) procedure that protects all suppressed cells in a single optimization?

A method that provides disclosure protection to all suppressed cells in a single step is a *multiple-cell complementary suppression* method. The existence of efficient multiple-cell methods is clouded by NP hardness results. Gusfield (1988) solved an important but restricted special case. A new method demonstrates that further progress is possible. The method is based on the following problem.

Problem. Given a single two-way table A and a set of suppressions S, select a minimal set of complementary suppressions so that each row and column of the table containing suppression(s) contains at least two suppressions.

Cox (1980) provided an optimal multiple-cell solution to this problem on which large-scale automated disclosure protection systems for the 1977 and 1982 U.S. Economic Censuses were based. The Problem is a necessary condition to the minimum-number-of-suppressions problem, but is not a sufficient condition in that there may exists patterns satisfying the two-or-more-suppressions-per-affected-row-or-column condition for which all suppressed cells are not contained in a cycle (i.e., fail the *cycle condition*). The number of patterns satisfying the two-or-more condition for a given table is computable and, in general, is large. In practice (e.g., U.S. Economic Censuses), most two-or-more patterns also satisfy the cycle condition, and in any case it usually is not difficult to augment one failing this condition. It is then reasonable to explore the Problem further.

The Cox (1980) solution to the Problem is a combinatorial procedure based on the notion that the best candidates for complementary suppressions are unsuppressed cells at the intersections of rows and columns requiring complementary suppression, the next-best candidates are cells at the intersection of rows or columns requiring complementary suppression with columns or rows containing suppressions, and the other cells are, in general, poorer candidates and will not lead to optimal solutions. This was formalized in a constructive mathematical theorem on which the procedure is based.

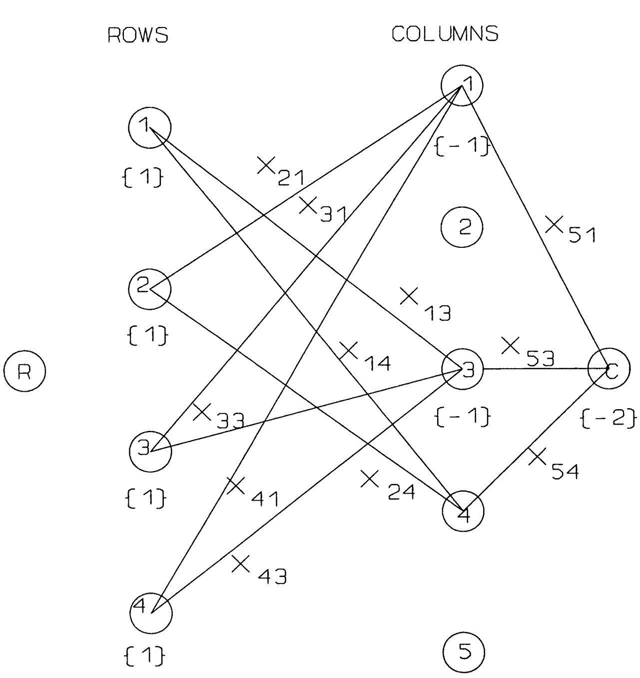

Figure 3. Network for Multiple-Cell Suppression in Table 1.

Theorem (Cox 1980). Let A be a general two-way table with suppressions under the minimum-number-of-complementary-suppressions criterion. Let m' (respectively, m'') denote the number of rows of $\mathbf{A}$ containing suppressions (respectively, requiring complementary suppression), and define n' (respectively, n'') similarly. Without loss of generality, assume that $m'' \geq n''$ and $m'' \geq 1$. If $\max\{m', n'\} = 1$, then the Problem can be solved by three complementary suppressions. Otherwise, m'' complementary suppressions are sufficient.

The Problem can be formulated as a network optimization problem (M, e), as follows. For clarity of presentation, it suffices to assume that suppression is limited to the internal entries of the table. There are two degenerate cases, each solvable by methods established earlier: $\max\{m', n'\} = 1$ and, $n'' = 0$ and $n' = 1$. The first case can be solved using the network model (N, c) of Section 4.1. The second case, which occurs when all suppressions occur in a single column J, can be solved using (N, c) with the exception: $c_{iJ} = -(c_0 + 1)$ for $(i, J) \in U$. The general case, $\min\{m', n'\} > 1$, is solved using the following network.

Network M. The node set of M equals the node set of N. The arc set of M consists of one arc from each row node $\mathbf{i}$ corresponding to a row requiring complementary suppression to each column node $\mathbf{j}$ corresponding to a column containing suppressions (flows denoted by $\mathbf{x}_{ij}$), and one arc from each of these column nodes to the second grand total node (flows denoted by $\mathbf{x}_{m+1,j}$). Arc capacities are $u_{m+1,j} = \infty, u_{ij} = 0$ if $(i, j) \in S$ or if $a_{ij} = 0$, and $u_{ij} = 1$ otherwise. Node requirements are zero, except the requirement of a row node corresponding to a row requiring complementary suppression equals $+1$, that of a column node corresponding to a column requiring complementary suppression equals -1, and that of the second grand total node equals $-(m'' - n'')$.

The node requirements ensure a total flow of m'' units, one unit along each row, corresponding to the minimum number of complementary suppressions m'' required to disclosure protect a general table—i.e., one in each row requiring complementary suppression. The column node requirements ensure that each column requiring complementary suppression receives at least one complementary suppression. The distribution of flow only to columns containing suppressions ensures that new column suppression problems are not created. Figure 3 illustrates the network M corresponding to the complementary suppression problem for Table 1. Node requirements are denoted by { }; zero-capacity arcs have been deleted.

In general tables, any basic feasible solution to M that obeys the cycle condition is an optimal solution to the multiple-cell minimum-number-of-complementary-suppressions problem. For positive tables, additional conditions often must be imposed. These conditions are expressible in terms of arc capacities u and a cost function e for M. To obtain a sufficient solution in one optimization (whenever such exists), the condition $u_{ij} = 0$ if $a_{ij} < \max\{p_{IJ}\colon (I, J) \in S\}$ is imposed. To include a secondary optimum (e.g., minimum-total-value-suppressed subject to minimum-number-of-complementary-suppressions), costs such as $e_{ij} = a_{ij}$ or based on (4) can be used. This two-stage optimization is the data loss criterion used for the 1977–1982 U.S. Economic Censuses. It was implemented using the theorem of Cox (1980) coupled with a branch-and-bound procedure. The implementation here, involving a single network optimization, is more efficient.

The imposition of additional conditions can produce an infeasible network (i.e., one corresponding to a suppression problem that cannot be solved completely in one iteration). In such cases it is worthwhile to identify and often to build iteratively on partial solutions. This can be accomplished by adding to M uncapacitated arcs of high cost from each row node corresponding to a row requiring complementary suppression to the first grand total node and similar arcs from the first grand total node to each column node corresponding to a column requiring complementary suppression. These arcs ensure a feasible network.

Partial suppression patterns obtained in this manner can be used in several ways. For example, complete the partial solution by another method; use the partial solution to identify high-priority candidates for complementary suppression; relax certain conditions by incorporating them into the cost function, as in (6); resort to appropriate single-cell procedures; and use an iterative procedure based on (M, e) analogous to IMP(I, J). Under the iteration option, an additional relaxation is available: add arcs of high cost and capacity equal to one to the remaining column nodes from the row nodes corresponding to rows requiring complementary suppression and similar arcs to these nodes from the second grand total node (recall that $m'' \geq n''$ to simplify notation). In many cases a solution can be obtained by means of this relaxation, followed by a second iteration to ensure at least two suppressions per newly affected column. Although not optimal in general, (M, e) is a good method to try first, because it is produces multiple-cell solutions.

(M, e) can be used to generate potential multiple-cell solutions in one optimization step. A resulting strategy to disclosure protect a table is to apply (M, e), with appropriate additional conditions, to create a (partial) pattern S, and then apply appropriate (N, c) iteratively to verify that S satisfies the cycle condition and, if it does not, to create an augmented pattern that does.

An optimal solution to the cell suppression problem of Table 1 consists of the following complementary suppressions: (1, 4), (2, 1), (3, 3), and (4, 1). This is a minimum-number-of-complementary-suppressions solution that provides sufficient protection to all table cells. It is also a minimum-total-value-suppressed solution. This "dual-optimum" solution is due only to contrivance in assigning the table values and is unlikely to occur in practice.

5. DISCUSSION

Protection intervals were assumed to be symmetric to simplify presentation. There are several ways to deal with nonsymmetric protection intervals using network optimization. The most direct way is to perform two optimizations per suppressed cell, one using N with p_{IJ} equal to the amount of upper protection required for (I, J) and the other using N with $u_{IJ-} = 1, u_{IJ+} = 0$, and p_{IJ} equal to the amount of lower protection required. The approach used for the U.S. Economic Censuses performs a single optimization with p_{IJ} equal to the maximum of these two values. A third method couples upper and lower protection networks and optimizes them simultaneously (Kelly, Golden, and Assad 1992), and can be used for the case of *variable protection bounds* where only the sum of the upper and lower protection is fixed.

The methods presented here for complementary cell suppression are of use to any organization that releases statistical data derived from confidential respondent information. The principal consumers and users of this methodology will be national statistical offices, whose dominant need centers on confidentiality protection in large-scale censuses and surveys. These needs are multifaceted, and thus it is unlikely that any one formulation will perform best in all situations. For example, in tables containing many zero cells (*sparse* tables), methods using stratified costs, as in (6), are likely to find solutions more efficiently than iterative methods such as IP(I, J). Conversely, in tables with few zeroes and in which the size distribution of cells is not extreme, heuristic methods that first seek an optimal solution (e.g., by setting $u_{ij*} = 0$ if $a_{ij} < p_{IJ}$) are likely to perform well.

Actual disclosure situations tend to be complex. For example, it may be necessary to prohibit the suppression of certain cells, such as important industry totals, or to encourage the suppression of other cells, such as values that are not shown in the published tables. Such *preference criteria* are easily met within the network optimization framework (e.g., by setting $u_{ij*} = 0$ or $c_{ij*} = 1$), but may unduly constrain the problem and, in large applications, become difficult to accommodate if extended to relative preferences, i.e., if it is more desirable to suppress cell (i, j) than to suppress cell (i^*, j^*). Also, cost functions for hierarchies can be considerably more complex than those for an individual table.

Another factor related to the applicability of network cell suppression methods in general situations are mathematical limitations on the class of linear aggregation systems that can be represented as networks. This class includes two-dimensional tables and hierarchies of two-dimensional tables, but excludes higher-dimensional tables and structures more complex than hierarchies (see Cox 1993 for details). Disclosure situations comprised predominantly of two-dimensional hierarchies are ideally suited to these methods; others may be only partially suited. Fortunately, most national censuses and surveys are comprised entirely or predominantly of two-dimensional hierarchies. In addition, research into *coupled* networks or networks embedded in linear programs may extend the applicability of these methods.

The methods presented here offer theoretical advantages over methods in current use (viz., formulation of a dichotomous decision problem and its cost function explicitly as a linear optimization problem), as well as practical advantages (viz., computational efficiency, reliance on standard methods and software, flexibility in use). The use of stratified cost functions enables finding optimal solutions and combining multiple optimizations into a single optimization. The objective of this article is to provide improved formulations and approaches for solving the complementary cell suppression problem, and to encourage their use. The use and extension of these methods (e.g., improved cost formulations or new formulations for other problems) would benefit from careful examination of their performance using real data in actual disclosure situations. Federal Committee on Statistical Methodology (1994) urged statistical agencies to compare confidentiality protection methods in these ways. Comparison of the performance of heuristic methods presented here for the minimum-total-value-suppressed problem in positive tables with other heuristic methods for this problem would be worthwhile.

[Received November 1993. Revised September 1994.]

REFERENCES

Carvalho, F. de, Dellaert, N., and Osorio, M. de Sanches (1994), "Statistical Disclosure in Two-Dimensional Tables: General Tables," *Journal of the American Statistical Association,* 89, 1547–1557.

Causey, B., Cox, L., and Ernst, L. (1985), "Applications of Transportation Theory to Statistical Problems," *Journal of the American Statistical Association,* 80, 392, 903–909.

Cox, L. (1980), "Suppression Methodology and Statistical Disclosure Control," *Journal of the American Statistical Association,* 75, 377–385.

——— (1981), "Linear Sensitivity Measures in Statistical Disclosure Control," *Journal of Statistical Planning and Inference,* 5, 153–164.

——— (1987), "New Results in Disclosure Avoidance for Tabulations," in *Proceedings of the 46*th *Session, International Statistical Institute,* Contributed Papers, pp. 83–84.

——— (1993), "Solving Confidentiality Problems in Tabulations Using Networks: A Network Model for Cell Suppression in U.S. Economic Censuses," in *Proceedings of the International Seminar on Statistical Confidentiality,* Dublin, EUROSTAT, pp. 229–245.

Cox, L., Fagan, J., Greenberg, B., and Hemmig, R. (1986), "Research at the Census Bureau Into Disclosure Avoidance Techniques for Tabular Data," in *Proceedings of the Survey and Research Methods Section, American Statistical Association,* pp. 388–393.

Cox, L., and George, J. (1989), "Controlled Rounding for Tables With Subtotals," *Annals of Operations Research,* 20, 141–157.

Federal Committee on Statistical Methodology (1994), *Report on Disclosure Limitation Methodology,* Statistical Policy Working Paper 22, U.S. Office of Management and Budget.

Glover, F., Klingman, D., and Phillips, N. (1992), *Network Models in Optimization and Their Applications in Practice,* New York: Wiley-Interscience.

Gusfield, D. (1988), "A Graph Theoretic Approach to Statistical Data Security," *SIAM Journal on Computing,* 17, 552–571.

Kelly, J., Golden, B., and Assad, A. (1992), "Cell Suppression: Disclosure Protection for Sensitive Tabular Data," *NETWORKS,* 22, 397–417.

Kennington, J., and Helgason, R. (1980), *Algorithms for Network Programming,* New York: John Wiley.

Robertson, D. (1993), "Cell Suppression at Statistics Canada," in *Proceedings of the Annual Research Conference, U.S. Bureau of the Census,* U.S. Department of Commerce, Washington, DC, pp. 107–131.