Partial Cell Suppression: a New methodology for Statistical Disclosure Control

Matteo Fischetti * Juan-José Salazar-González †

October, 1998; revised, September 1999

Abstract

In this paper we address the problem of protecting confidentiality in statistical tables containing sensitive information that cannot be disseminated. This is an issue of primary importance in practice. Cell Suppression is a widely-used technique for avoiding disclosure of sensitive information, which consists in suppressing (i.e., replacing by "missing") all sensitive table entries along with a certain number of other entries, called complementary suppressions. Determining a pattern of complementary suppressions that minimizes the overall loss of information results into a difficult (i.e., $\mathcal{NP}$-hard) optimization problem known as the Cell Suppression Problem.

We propose here a different protection methodology consisting of replacing some table entries by appropriate intervals containing the actual value of the unpublished cells. We call this methodology Partial Cell Suppression, as opposed to the classical "complete" (i.e., yes/no) cell suppression. Partial suppression has the important advantage of reducing the overall information loss needed to protect the sensitive information. Also, the new method provides automatically auditing ranges for each unpublished cell, thus saving an often time-consuming task to the statistical office while increasing the information explicitly provided with the table. Moreover, we propose an efficient (i.e., polynomial-time) algorithm to find an optimal partial suppression solution. A preliminary computational comparison between partial and complete suppression methodologies is reported. Finally, we address possible extensions leading to a unified complete/partial cell suppression framework.

Keywords: Statistical Disclosure Control, Cell Suppression, Linear Programming.

*DEI, University of Padova, Italy; fisch@dei.unipd.it

†DEIOC, University of La Laguna, Spain; jjsalaza@ull.es

	A	B	C	Total
Activity I	20	50	10	80
Activity II	8	19	**22**	49
Activity III	17	32	12	61
Total	45	101	44	190

Figure 1: Investment of enterprises by activity and region.

1 Introduction

A statistical agency collects data to be processed and published. This data is often obtained from individual respondents under a pledge of confidentiality: statistical agencies have the responsibility of not releasing any data or data summaries from which individual respondent information can be revealed (*sensitive data*). On the other hand, statistical agencies aim at publishing as much information as possible. This results into a trade-off between privacy rights and information loss, an issue of primary importance in practice. The interested reader is referred to Willenborg and De Waal [17] for a discussion on disclosure control methodologies.

Cell suppression is one of the most popular techniques for disclosure avoidance, and is typically applied to 2- or 3-dimensional tables whose entries (*cells*) are subject to marginal totals. We next outline the standard *complete* cell suppression technique, and then introduce an alternative version that we call *partial* cell suppression. Both methodologies are illustrated by means of the following introductory example, taken from Willenborg and De Waal [17].

Figure 1 exhibits a table giving the investment of enterprises (per millions of guilders), classified by activity and region. Let us assume that the information in the cell corresponding to Activity II and Region C is confidential, hence it is viewed as a *sensitive cell* to be suppressed. By using the marginal totals an *attacker* interested in the disclosure of the sensitive cell can however recompute its missing value, hence other table entries must be suppressed as well, e.g., those of Figure 2.

With this choice, the attacker cannot disclose the nominal value of the sensitive cell exactly, although he/she can still compute a range for the values of this cell which are consistent with the published entries. Indeed, the minimum value y_{23}^{-} for sensitive cell (2,3) can be computed by solving a linear program in which the values y_{ij} for the

	A	B	C	Total
Activity I	20	50	10	80
Activity II	*	19	*	49
Activity III	*	32	*	61
Total	45	101	44	190

Figure 2: A possible complete CSP output

missing cells (i, j) are treated as unknowns, namely

$$y_{23}^{-} := \min y_{23}$$

subject to

$$\begin{array}{rrrrl} y_{21} & & +y_{23} & & = 30 \\ & y_{31} & & +y_{33} & = 29 \\ y_{21} & +y_{31} & & & = 25 \\ & & y_{23} & +y_{33} & = 34 \\ y_{21}, & y_{23}, & y_{31}, & y_{33} & \geq 0. \end{array}$$

Notice that the right-hand side values are known to the attacker, as they can be obtained as the difference between the marginal and the published values in a row/column.

The maximum value y_{23}^{+} for the sensitive cell can be computed in a perfectly analogous way, by solving the linear program of maximizing y_{23} subject to the same constraints as before.

In the example, $y_{23}^{-} = 5$ and $y_{23}^{+} = 30$, i.e., the sensitive information is "protected" within the *protection interval* [5,30]. If this interval is considered sufficiently wide by the statistical office, then the sensitive cell is called *protected*; otherwise new complementary suppressions are needed. Notice however that the extreme values of interval $[5, 30]$ are only attained if the cell corresponding to Activity II and Region A takes the quite unreasonable values of 0 and 25. Bounding the cell variation to $\pm 50\%$ of the nominal value (say) results in the more realistic protection interval $[18, 26]$.

The statistical office then aims at finding a set of complementary suppressions protecting all the sensitive cells against an attacker, and such that the loss of information associated with the suppressed entries is minimized.

This combinatorial optimization problem is known as the *(Complete) Cell Suppression Problem*, or CSP for short. CSP belongs to the class of the strongly $\mathcal{NP}$-hard problems (see, e.g., Kelly, Golden and Assad [13], Geurts [9], Kao [11]), meaning that

it is very unlikely that an algorithm for the exact solution of CSP exists, which guarantees an efficient (i.e., polynomial-time) performance for all possible input instances.

Previous work on CSP from the literature mainly concentrate on 2-dimensional tables with marginals. Heuristic solution procedures have been proposed by several authors, including Cox [1, 2], Sande [16], Kelly, Golden and Assad [13], and Carvalho, Dellaert and Osório [4]. Kelly [12] proposed a mixed-integer linear programming formulation involving a huge number of variables and constraints. Geurts [9] refined this model, and reported computational experiences on small-size instances, the largest instance solved to optimality being a table with 20 rows, 6 columns and 17 sensitive cells. Gusfield [10] gave a polynomial algorithm for a special case of the problem. Heuristics for 3-dimensional tables have been proposed in Robertson [15], Sande [16], and Dellaert and Luijten [5]. Very recently, we proposed in [7] a new method capable of solving to proven optimality, on a personal computer, 2-dimensional tables with about 250,000 cells and 10,000 primary suppressions. An extension of this methodology resulted into an algorithm presented in [6], capable of solving to proven optimality real-world 3- and 4-dimensional tables as well as linked tables of reasonable size.

In this paper we address cell suppression in the very general context of tables whose entries are linked by a generic system of linear equations, as we proposed in [6]. However, we introduce and analyze here an alternative protection methodology that we call *partial cell suppression*. The methodology consists in publishing intervals for some table entries, as opposed to replacing some table entries by "missing". Figure 3 exhibits a possible partial cell suppression output for the original table in Figure 1, to be compared with the complete cell suppression output in Figure 2.

The new method has important advantages over complete (i.e., standard) cell suppression.

A first advantage is related to computational complexity, in that an optimal partial suppression solution can be found efficiently, i.e., in polynomial time. In this view, large hierarchical and linked tables can hopefully be optimized in reasonable computing time.

A second advantage is that the methodology itself provides, with no extra computational effort, the *auditing range* **for each partially suppressed entry, defined by the minumum and maximum cell values that an attacker can compute by using the published information. Hence the published table entries are considerably more informative than the "emply" cells**

	A	B	C	Total
Activity I	[18,24]	50	[6,12]	80
Activity II	[4,10]	19	[20,26]	49
Activity III	17	32	12	61
Total	45	101	44	190

Figure 3: A possible partial CSP output

which are the output of the complete cell suppression methodology.

The third (and most important) advantage is related to the the loss of information needed to meet the protection levels required by the statistical office, in that partial cell suppression has a complete freedom in "tailoring" the published cell ranges so as to avoid wasting information associated with the suppressed cells. This behavior will become evident in the computational section, where we compare complete and partial cell suppression methodologies on three real-world instances. In this respect, partial cell suppression compares favorably also with other (more rigid) perturbation techniques such as *Controlled Rounding*, **where each table entry is in a sense replaced by an interval centered at the published rounded value, the interval width being fixed** *a prori* **to approximately twice the rounding base (e.g., assuming a rounding base of 10, a published rounded value of 50 actually represents the interval** $[41, 59]$**). A similar reasoning applies when constrasting partial cell suppression with** *Source Data Perturbation*, **a fairly easy-to-implement methodology in which the base file (microdata) is perturbed so as to provide automatically protection for all tabulations derived from the file. Indeed, the amount of perturbation to be added to the base file cannot be tuned as finely as in the case of partial cell suppression, which may results into a large information loss of the tabular data due to overperturbation of the microdata.**

The paper is organized as follows. Section 2 gives a formal description of the problem we are dealing with. **Our mathematical model is outlined in Section 3, where we also discuss briefly a possible solution method based on run-time cutting plane generation. Preliminary computational results are given in Section 4, where we compare the partial and complete cell suppression methodologies on real-world test instances. Some conclusion are finally**

drawn in Section 5, where we also sketch a unifying framework for partial and complete suppression.

A preliminary version of the present paper was presented by the authors at the *International Symposium on Linked Employer-Employee Data*, Washington, May 21–22, 1998.

2 Partial vs. Complete Cell Suppression

We first give a formal definition of the complete (i.e., yes/no) cell suppression problem, in the general form we introduced in [6]. A *table* is a vector $a = [a_1, \ldots, a_n]$ satisfying a given set of linear constraints known to a possible attacker, namely

$$My = b \tag{1}$$

$$lb_i \leq y_i \leq ub_i \quad \text{for all } i = 1, \ldots, n. \tag{2}$$

In other words, system (1)–(2) models the whole *a priori* information on the table known to an attacker, and $y := a$ is a possible solution. Typically, each equation in (1) corresponds to a marginal entry, whereas inequalities (2) enforce the public "external bounds". In the case of k-dimensional tables with marginals, each equation in (1) is of the type $\sum_{j \in Q_i} y_j - y_i = 0$, where index i corresponds to a marginal entry and Q_i to the associated set of internal table entries.

A *cell* is an index corresponding to an entry of the table. Given a *nominal table* a, let $\{i_1, \ldots, i_p\}$ denote the set of sensitive cells in a to be protected, as identified by the statistical office according to some criteria.

A *complete suppression pattern* is a subset of cells $SUP \subseteq \{1, \ldots, n\}$ corresponding to the unpublished cells. A *consistent table* with respect to a given suppression pattern SUP and to a given nominal table a is a vector $y = [y_1, \ldots, y_n]$ satisfying

$$\begin{aligned} My &= b \\ lb_i \leq y_i &\leq ub_i \qquad \text{for all } i \in SUP \\ y_i &= a_i \qquad \text{for all } i \notin SUP \end{aligned}$$

where the latter equations impose that the components of y associated with the published entries coincide with the nominal ones. In other words, any consistent table gives a feasible way the attacker can fill the missing entries of the published table.

For each sensitive cell i_k $(k = 1, \ldots, p)$, the statistical office provides three nonnegative values, namely LPL_{i_k}, UPL_{i_k} and SPL_{i_k}, called *Lower Protection Level*, *Upper Protection Level* and *Sliding Protection Level*, respectively. In typical applications, these values are computed as a certain percentage of the nominal value a_{i_k}. A complete suppression pattern is therefore considered *feasible* by the statistical office if it guarantees the required protection intervals against an attacker, in the sense that, for each sensitive cell i_k $(k = 1, \ldots, p)$, there exist two feasible tables, say f^k and g^k, such that $f^k_{i_k} \leq a_{i_k} - LPL_{i_k}$, $g^k_{i_k} \geq a_{i_k} + UPL_{i_k}$, and $g^k_{i_k} - f^k_{i_k} \geq SPL_{i_k}$. In other words, it is required that $y^-_{i_k} \leq a_{i_k} - LPL_{i_k}$, $y^+_{i_k} \geq a_{i_k} + UPL_{i_k}$, and $y^+_{i_k} - y^-_{i_k} \geq SPL_{i_k}$, where

$$\left.\begin{array}{rrcll} y^-_{i_k} := & \min y_{i_k} & & & \\ & My & = & b & \\ & lb_i \leq y_i & \leq & ub_i & \text{for all } i \in SUP \\ & y_i & = & a_i & \text{for all } i \notin SUP \end{array}\right\}$$

and

$$\left.\begin{array}{rrcll} y^+_{i_k} := & \max y_{i_k} & & & \\ & My & = & b & \\ & lb_i \leq y_i & \leq & ub_i & \text{for all } i \in SUP \\ & y_i & = & a_i & \text{for all } i \notin SUP \end{array}\right\}$$

As already mentioned, the statistical office is interested in selecting, among all complete suppression patterns, a feasible one with minimum information loss. This issue can be modeled by associating a *weight* $w_i \geq 0$ with each cell i of the table, and by requiring the minimization of the overall weight of the suppressed cells, computed as $\sum_{i \in SUP} w_i$ (in typical applications, the weights w_i provided by the statistical office are proportional to a_i or to $\log a_i$). The resulting combinatorial problem will be denoted as the *Complete Cell Suppression Problem*, or complete CSP for short.

We next introduce **our** new cell suppression methodology, leading to what we call the *Partial CSP*. As already mentioned, the idea is to allow each table entry a_i to be replaced by a convenient interval $[a_i - z^-_i, a_i + z^+_i]$ (case $z^-_i = z^-_i = 0$ being allowed). The extreme values of each interval have then to be determined so as to ensure the required protection for the sensitive entries, while minimizing the overall loss of information incurred.

More specifically, assume that the statistical office is willing to publish, for each cell i, an interval $[a_i - z^-_i, a_i + z^+_i]$, where $a_i - z^-_i \geq lb_i$ and $a_i + z^+_i \leq ub_i$ for consistency with the *a priori* information known to the attacker, and $z^-_i \geq 0$ and $z^+_i \geq 0$ so as

to avoid publishing wrong information. **Given such intervals, an attacker can then compute the** *auditing range* $[y_{i_k}^-, y_{i_k}^+]$ **containing possible values for each sensitive cell** i_k**, by solving the following two LP's:**

$$\left.\begin{array}{rrcll} y_{i_k}^- := & \min y_{i_k} & & & \\ & My & = & b & \\ & a_i - z_i^- \leq y_i & \leq & a_i + z_i^+ & \text{for all } i = 1, \ldots, n \end{array}\right\}$$

and

$$\left.\begin{array}{rrcll} y_{i_k}^+ := & \max y_{i_k} & & & \\ & My & = & b & \\ & a_i - z_i^- \leq y_i & \leq & a_i + z_i^+ & \text{for all } i = 1, \ldots, n \end{array}\right\}$$

As in the case of complete CSP, the statistical office is interested in defining the interval $[a_i - z_i^-, a_i + z_i^+]$ **for each cell** i **so as to guarantee the required protection of the sensitive entries against a possible attacker.** In other words, for each sensitive cell i_k we must have

$$\begin{array}{rl} \text{lower protection level requirement:} & y_{i_k}^- \leq a_{i_k} - LPL_{i_k} \\ \text{upper protection level requirement:} & y_{i_k}^+ \geq a_{i_k} + UPL_{i_k} \\ \text{sliding protection level requirement:} & y_{i_k}^+ - y_{i_k}^- \geq SPL_{i_k} \end{array}$$

In particular, this implies $z_{i_k}^+ \geq UPL_{i_k}$, $z_{i_k}^- \geq LPL_{i_k}$, and $z_{i_k}^+ + z_{i_k}^- \geq SPL_{i_k}$ for all sensitive cells i_k.

Moreover, the published intervals should be as narrow as possible so as to minimize the overall loss information, computed as

$$\sum_{i=1}^{n} (w_i^+ z_i^+ + w_i^- z_i^-)$$

for some pre-specified nonnegative weights w_i^+ and w_i^-. For example, setting $w_i^+ = w_i^- = 1$ for all i leads to the figure $\sum_{i=1}^{n}(z_i^+ + z_i^-)$, i.e., the overall loss of information equals to the sum of the interval widths.

An important observation is that the nonnegativity assumption on the weights implies minimality of the ranges $[a_i - \tilde{z}_i^-, a_i + \tilde{z}_i^+]$ associated with an optimal solution $(\tilde{z}^+, \tilde{z}^-)$, i.e., these ranges actually coincide with the auditing ranges $[y_i^-, y_i^+]$ an attacker can recompute, for each cell i, from the published information.

3 A Linear Programming model for partial CSP

We next introduce **a Linear Programming (LP) model for the partial CSP defined in the previous section. The model is** in the spirit of the Mixed-Integer LP model we gave in [6] for complete CSP, but it does not need any 0-1 variable to model the yes/no nature of a complete suppression pattern.

For notational convenience, let us define the relative external bounds $LB_i := a_i - lb_i \geq 0$ and $UB_i := ub_i - a_i \geq 0$, i.e., the *a priori* range of feasible values for cell i known to the attacker is $[a_i - LB_i, a_i + UB_i]$. We also define $LPL_i := UPL_i := SPL_i := 0$ for all nonsensitive cells i, and assume, for consistency reasons, that $UPL_i \leq UB_i$, $LPL_i \leq LB_i$, and $SPL_i \leq LB_i + UB_i$ for all i.

Our LP model for partial CSP requires the introduction, for each $k = 1, \ldots, p$, of auxiliary continuous variables $f^k = [f_1^k, \ldots, f_n^k]$ and $g^k = [g_1^k, \ldots, g_n^k]$ defining tables which are consistent with the published intervals, and certificating the fulfillment of the protection level requirements. The model then reads:

$$\min \sum_{i=1}^{n} (w_i^+ z_i^+ + w_i^- z_i^-) \tag{3}$$

subject to

$$UPL_i \leq z_i^+ \leq UB_i \qquad \text{for } i = 1, \ldots, n \tag{4}$$

$$LPL_i \leq z_i^- \leq LB_i \qquad \text{for } i = 1, \ldots, n \tag{5}$$

$$SPL_{i_k} \leq z_{i_k}^+ + z_{i_k}^- \qquad \text{for } k = 1, \ldots, p \tag{6}$$

and, for each sensitive cell i_k $(k = 1, \ldots, p)$:

$$\left.\begin{array}{rrll} M & f^k & = b & \\ a_i - z_i^- \leq & f_i^k & \leq a_i + z_i^+ & \text{for } i = 1, \ldots, n \\ & f_{i_k}^k & = a_{i_k} - z_{i_k}^- & \end{array}\right\} \tag{7}$$

$$\left.\begin{array}{rrll} M & g^k & = b & \\ a_i - z_i^- \leq & g_i^k & \leq a_i + z_i^+ & \text{for } i = 1, \ldots, n \\ & g_{i_k}^k & = a_{i_k} + z_{i_k}^+ & \end{array}\right\} \tag{8}$$

Constraints (4)–(6) express the consistency of the published intervals $[a_i - z_i^-, a_i + z_i^+]$ with the external bounds known to an attacker, and also enforce the protection levels. Constraints (7) and (8) require both extreme values of the published intervals

for sensitive cells be attained in some tables which are consistent with the published one. Similar constraints for nonsensitive cells could be considered as well, but they are redundant under the weight nonnegativity assumption.

The solution of the LP model (3)–(8) can be obtained in polynomial time, as it involves a number of variable and constraints which is polynomially bounded in the input size. In practice, however, the model cannot be handled effectively, even for very small instances, due to the really huge number of auxiliary variables f_i^k and g_i^k and of linking constraints between the (z^+, z^-) and the auxiliary variables. For example, for a 100×100 table with marginals having 5% sensitive cells the model would need more than 10,000,000 variables and 20,000,000 constraints, a size that cannot be handled explicitly by the LP technology of today.

However, an equivalent model inspired by Benders' decomposition (see e.g. [14]) can be derived, which is based on the same technique we used in [6] in the context of complete CSP.

The idea is to use standard LP duality theory to project onto the (z^+, z^-)-subspace the polyhedron containing the feasible solutions of model (3)–(8), thus avoiding the explicit introduction of the auxiliary variables f^k and g^k $(k = 1, \ldots, p)$ along with the associated linking constraints. In the new model, the protection level requirements are in fact imposed by means of families of linear constraints (called *capacity constraints*) in the space of the (z^+, z^-)-variables only. Although the number of potential capacity constraints is in principle exponential in the input size, for a given partial CSP instance just a few such constraints turn out to be relevant, all the others being redundant. Unfortunately, we cannot decide in advance which capacity constraints are relevant for a given input instance. Hence we apply a dynamic constraint-generation technique that identifies them "on the fly", during the solution of a relaxed LP problem, according to the *cutting-plane* scheme sketched below; the interested reader is referred to [8] for more details.

We start by solving the so-called *master LP*:

$$\min\{\sum_{i=1}^{n}(w_i^+ z_i^+ + w_i^- z_i^-) : \ \textbf{(4)–(6) hold}\},$$

in which we only impose the trivial consistency conditions on the z_i^+ and

z_i^- **variables. Let** $(\tilde{z}^+, \tilde{z}^-)$ **be the optimal solution found. Our order of business is to check whether the vector** $(\tilde{z}^+, \tilde{z}^-)$**, now viewed as a given parameter, guarantees the required protection levels. In geometrical terms, this is equivalent to finding a hyperplane in the** (z^+, z^-)**-space that separates** $(\tilde{z}^+, \tilde{z}^-)$ **from the polyhedron defined by the capacity constraints. The resulting** *separation problem* **amounts to the solution of a series of (small size) LP problems [8], hence it typically requires very short computing time. All the capacity constraints violated by** $(\tilde{z}^+, \tilde{z}^-)$ **that are detected in this way, if any, are added to the current master problem, and the procedure is iterated by re-optimizing the new master problem and by re-entering the separation procedure. If no violated capacity constraint is found, instead, then the optimal solution** $(\tilde{z}^+, \tilde{z}^-)$ **of the current master problem is guaranteed to be an optimal solution of LP model (3)–(8) .**

4 Computational results

A very preliminary implementation of the cutting-plane solution algorithm has been obtained along the following lines. We eliminated the two decision vectors $z^+, z^- \in I\!R^n$ by introducing a single decision vector $x \in I\!R^n$ such that

$$\left.\begin{array}{c} z_i^+ = UB_i\, x_i \\ z_i^- = LB_i\, x_i \\ 0 \le x_i \le 1 \end{array}\right\} \qquad i = 1, \ldots, n.$$

This is equivalent to force variables z_i^+ and z_i^- to vary according to the prefixed ratio given by $LB_i z_i^+ = UB_i z_i^-$. As a consequence, the new model has fewer x-variables, but its optimal solution value can be strictly worse than in the original model.

The resulting simplified model was then solved by modifying slightly the code we presented in [6] for complete CSP. In fact, it was sufficient to deactivate all the "model improvements" related to the integrality of the x-variables, as described in Section 4.2 of [6], and to stop code execution right after the processing of the root node of the enumerative tree; see [6] for details.

The modified software was compiled with Watcom C/C++ 10.6 and run on a PC Pentium/133 with 32 RAM Mbytes under Windows 95. As to the LP-solver, we used the commercial package CPLEX 3.0. Our preliminary test bed consists of the 2-dimensional real-world instances addresses in [6].

name	type	cells	links	p	pl
CBS1	41×31	1271	72	3	6
CBS2	183×61	11163	244	2467	4934
CCSR	359×46	16514	405	4923	9846

Table 1: Real-world instances.

Table 1 reports information about the test bed. For each instance, the table gives:

name: name of the instance;

type: table size;

cells: number of cells in the table (=n);

links: number of equations in the table (=number of rows of matrix M);

p: number of sensitive cells (primary suppressions);

pl: number of nonzero protection levels before problem reduction;

As to the critical values LB_i and UB_i defining the *a priori* intervals known to an attacker, we set $UB_i := a_i$ and $LB_i := a_i/2$ for each $i = 1, \ldots, n$, i.e., we assumed each table entry i is known with $[+100\%, -50\%]$ uncertainty. Protection levels were set to $UPL_i := a_i/2$ and $LPL_i := a_i/4$ for each $i = 1, \ldots, n$, thus requiring 75% protection levels for the sensitive entries. All fractional input values were rounded up to the nearest integer.

For each cell i, the partial suppression weights w_i^+ and w_i^- have been set to 1, whereas the weight for complete suppression has been defined as $w_i = UB_i + LB_i$, meaning that any completely suppressed cell introduces a loss of information equal to the width of the *a priori* interval known to the attacker for the missing value a_i. This produces complete CSP instances which can fairly be compared with their corresponding partial CSP versions.

Table 2 gives the following information.

name: name of the instance;

sup: number of partial suppressions in an optimal partial CSP solution;

loss: total loss of information in an optimal partial CSP solution;

name	sup	loss	time	sup'	loss'	time'	time"
CBS1	9	153	4.78	7	486	367.62	0.07
CBS2	6	140,297	17.28	3	281,398	11.42	667.06
CCSR	85	9,357	77.67	38	19,099	116.93	2486.07

Table 2: Partial vs. complete suppression.

time: overall computing time spent by our partial CSP approach (in Pentium/75 seconds);

sup': number of complete suppressions in an optimal complete CSP solution;

loss': total loss of information in an optimal complete CSP solution;

time': overall computing time spent by our complete CSP approach (in Pentium/75 seconds);

time": additional computing time needed to compute the auditing intervals $[y_i^-, y_i^+]$ for the unpublished cells in the optimal complete CSP solution.

The table clearly shows the information-loss savings obtainable by using the partial CSP methodology, even in the oversimplified version we implemented.

5 Conclusions and extensions

Cell suppression is a widely-used methodology in statistical disclosure control. In this paper we have introduced a different protection methodology consisting of replacing some table entries by appropriate intervals containing the actual value of the unpublished cells.

The new methodology, called partial cell suppression, has important advantages with respect to the classical one. First of all, it allows the statistical office to obtain the required protection levels with significantly smaller loss of information. In addition, the computational complexity of the new methodology is just polynomial as opposed to the exponential complexity of the classical cell suppression. Finally, the new method provides automatically auditing ranges for each unpublished cell, thus saving an often time-consuming task to the statistical office while increasing the information explicitly provided with the table.

We have also presented some preliminary computational results on a slight modification of our model.

Future work on the subject should address the issue of avoiding too many entries being replaced by intervals, even if this would produce a small increase of the overall information loss incurred. A possible solution is to associate a fixed penalty $w_i \geq 0$ to be applied whenever a table entry a_i is replaced by an interval, no matter how narrow the internal is. This requires the introduction in our model of 0-1 variables x_i $(i = 1, \ldots, n)$, where $x_i = 1$ means that the entry a_i is replaced by an interval $[a_i - z_i^-, a_i + z_i^+]$ with $z_i^+ + z_i^- > 0$. In this way, the objective function becomes

$$\sum_{i=1}^{n} (w_i^+ z_i^+ + w_i^- z_i^- + w_i x_i)$$

and can incorporate the yes/no suppression penalties.

Of course, the new variables have to be linked to the continuous variables (z^+, z^-) by the consistency constraints:

$$\left.\begin{array}{c} z_i^+ \leq UB_i \; x_i \\ z_i^- \leq LB_i \; x_i \\ x_i \in \{0,1\} \end{array}\right\} \qquad \text{for } i = 1, \ldots, n.$$

which enforce $x_i = 1$ whenever $z_i^+ > 0$ and/or $z_i^- > 0$, i.e., when a proper interval is published instead of a_i. Notice that, in our second model, the above constraints can dealt with at the master level only, hence they do not affect capacity constraint separation. Also to be incorporated at the master level are possible constraints of the type $\sum_{i=1}^{n} x_i \leq S_{max}$, imposing a maximum number S_{max} of "partially suppressed" entries, or constraints of the form $z_i^+ + z_i^- \geq W_i^{min} \; x_i$, imposing a minimum width W_i^{min} for each partially suppressed entry a_i.

The above extended model then generalizes and unifies both complete and partial cell suppression, and allows the user to have a complete control on the number of entries to be replaced by intervals as well as on the interval widths.

The exact/heuristic solution of the resulting $\mathcal{NP}$-hard optimization problem can be obtained by using enumeration techniques akin to the *branch-and-cut* method we described in [6] for complete cell suppression.

6 Acknowledgment

The first author was supported by *M.U.R.S.T., Ministero della Ricerca Scientifica e Tecnologica*, Italy, while the second author was supported by *Ministerio de Educacion y Ciencia*, Spain.

We thank Larry Cox (*U.S. Environment Protection Agency*, U.S.A.), Ramesh Dandekar (*U.S. Department of Energy*, U.S.A.), and Leon Willenborg (*Central Bureau of Statistics*, The Netherlands) for stimulating discussions on statistical disclosure control methodologies.

References

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

[2] L.H. Cox, "Network Models for Complementary Cell Suppression", *Journal of the American Statistical Association*, 90 (1995) 1453–1462.

[3] H.P. Crowder, E.L. Johnson, and M.W. Padberg, "Solving Large-Scale Zero-One Linear Programming Problems", *Operations Research*, 31 (1983) 803–834.

[4] F.D. Carvalho, N.P. Dellaert, M.S. Osório, "Statistical Disclosure in Two-Dimensional Tables: General Tables", *Journal of the American Statistical Association*, 89 (1994) 1547–1557.

[5] N. P. Dellaert, W. A. Luijten, "Statistical Disclosure in General Three-Dimensional Tables", Technical Paper TI 96-114/9, Tinbergen Institute (1996).

[6] M. Fischetti, J.J. Salazar, "Models and Algorithms for Optimizing Cell Suppression Problem in Tabular Data with Linear Constraints", **working paper ????????** , 1998.

[7] M. Fischetti, J.J. Salazar, "Models and Algorithms for the 2-Dimensional Cell Suppression Problem in Statistical Disclosure Control", *Mathematical Programming* **84, 283-312 (1999)**.

[8] **M. Fischetti, J.J. Salazar, "Models and Algorithms for Partial Cell Suppression in Statistical Disclosure Control", working paper ????? 1999, available at www.??????**

[9] J. Geurts, "Heuristics for Cell Suppression in Tables", working paper, Netherlands Central Bureau of Statistics, Voorburg, 1992.

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

[11] M.Y. Kao, "Data Security Equals Graph Connectivity", *SIAM Journal on Discrete Mathematics*, 9 (1996) 87–100.

[12] J.P. Kelly, "Confidentiality Protection in Two and Three-Dimensional Tables", Ph.D. dissertation, University of Maryland, College Park, Maryland, 1990.

[13] J.P. Kelly, B.L. Golden, and A.A. Assad, "Cell Suppression: Disclosure Protection for Sensitive Tabular Data", *Networks* 22 (1992) 397–417.

[14] G.L. Nemhauser, L.A. Wolsey, "Integer and Combinatorial Optimization, John Wiley & Sons, 1988.

[15] D.A. Robertson, "Cell Suppression at Statistics Canada", Proceedings of the *Second International Conference on Statistical Confidentiality*, Luxembourg, 1995.

[16] G. Sande, "Automated cell suppression to preserve confidentiality of business statistics", *Statistical Journal of the United Nations ECE* 2 (1984) 33–41.

[17] L.C.R.J. Willenborg, T. De Waal, "Statistical Disclosure Control in Practice", *Lecture Notes in Statistics 111*, *Springer*, New York, 1996.