SORT 35 (1) January-June 2011, 3-20

Extending controlled tabular adjustment for non-additive tabular data with negative protection levels

Jordi Castro*
Universitat Politècnica de Catalunya

Abstract

Minimum distance controlled tabular adjustment (CTA) is a recent perturbative methodology for the protection of tabular data. An implementation of CTA was recently used by Eurostat for the protection of European Union level structural business and animal production statistics. The real-world instances to be solved forced the classical CTA model to be extended with two features: first, to deal with non-additive tables; second, and most important, to consider negative protection levels. The latter extension means a significant modification of the classical CTA mixed integer linear model. We present and compare new models for these extensions. Computational results are reported using a set of real-world instances, and two state-of-the-art commercial solvers (CPLEX and Xpress).

MSC: 90C11, 90C06, 90B99, 62Q05

Keywords: Mathematical programming, mixed integer linear optimization, statistical disclosure control, perturbative methods, controlled tabular adjustment, official statistics

1. Introduction

Tabular data protection is one of the two disciplines of the statistical disclosure control field (microdata protection being the second one). The interested reader is addressed to the recent research monographs Willenborg and de Waal (2000); Domingo-Ferrer and Franconi (2006); Domingo-Ferrer and Franconi (2008) for an overview of this field. Controlled tabular adjustment (CTA) and other minimum distance related variants were suggested in Dandekar and Cox (2002) and Castro (2006) as a replacement to other

*Dept. of Statistics and Operations Research. Universitat Politècnica de Catalunya. Jordi Girona 1–3, 08034 Barcelona. jordi.castro@upc.edu. http://www-eio.upc.es/~jcastro

Received: April 2010
Accepted: November 2010

Table 1: *(a) Sizes of optimization problems associated to cell suppression (CSP), controlled rounding (CRP) and CTA. (b) Figures for a particular table of 4000 cells, 1000 sensitive cells, and 2500 linear relations.*

Problem	constraints	continuous	binary
CSP/CRP	$2(m+2n)s$	$2ns$	n
CTA	$m+4s$	$2n$	s

(a)

Problem	constraints	continuous	binary
CSP/CRP	21,000,000	8,000,000	4,000
CTA	6,500	8,000	1,000

(b)

computationally more expensive approaches for tabular data protection. CTA can be seen as a method for generating a safe synthetic table, which is as close as possible to the original table. This is obtained by solving the following optimization problem: given a non-safe table, with a set of sensitive cells to be protected, find the closest safe table to the original one (according to some distance) by adding the minimum amount of perturbations. Some of the good properties of CTA are:

- It can be applied to any table or set of linked tables. Even for complex and large tables a solution can be obtained in reasonable time (likely suboptimal, but with an acceptable optimality gap).
- From a computational point of view, the size of the resulting optimization problem is by far lower than for other well-known protection methods, such as the cell suppression problem (CSP) (Castro (2007a)) and the controlled rounding problem (CRP) (Salazar-González (2006)). For a table of n cells, s of them being sensitive, and m table linear relations, Table 1*(a)* shows the dimensions of the optimization problem formulated by CSP, CRP and CTA (number of constraints, and number of continuous and binary variables). For example, the particular figures for a table of 4000 cells, 1000 sensitive cells, and 2500 linear relations are provided in Table 1*(b)*, clearly showing the different order of magnitude between the optimization problems.
- State-of-the-art solvers, such as CPLEX (IBM ILOG CPLEX (2009)) or Xpress (FICO Dash Xpress (2008)), can be applied to the solution of CTA (at least for medium size instances). Other approaches like CSP or CRP require specialized solution methods, either optimal or heuristic, even for small instances. For very large-scale instances, it is possible to develop specialized, hopefully more efficient, procedures for CTA. Some preliminary work has already been started (Castro and Baena (2008), González and Castro (2009)), but they are beyond the scope of this work.

- Either L_1, L_∞ or Euclidean L_2 distances can be used in the objective function of CTA. L_2 distances provide mixed integer quadratic problems, which are more difficult to be solved, but reduce the largest deviations. L_1 provides simpler optimization problems, and it is currently mostly used by National Statistical Institutes. All the models in this paper use L_1.
- The particular model of CTA with the L_1 distance does not guarantee integrality of the perturbations (i.e., they can be fractional values); models with other distances (L_2 or L_∞) neither guarantee integrality. Indeed, it is possible to obtain tables where the perturbations are fractional (e.g., three-dimensional tables are modeled as a multicommodity flow problem (Castro (2005, 2007b)), which is known not to provide integral flows). However, in most tables tested with the L_1 distance, the solution provided was integer without imposing integrality of perturbations (however, we do not claim the matrices were totally unimodular, which is sufficient for guaranteeing integrality). Even if perturbations were not integer, they would still be valid for magnitude tables.
- Previous empirical testing (Castro and Giessing (2006)) showed the quality of the solution (measured as number of cells with large significant deviations) provided by CTA was comparable, even higher, than that obtained with CSP. Other quality criteria (Cox, Kelly and Patil (2004)) can also be easily added to the CTA formulation.

A package implementing CTA (Castro, González and Baena (2009)) has recently been incorporated within a wider scheme for the protection of structural business statistics disseminated by Eurostat (project coordinated by Statistics Netherlands, with the participation of Destatis and Universitat Politècnica de Catalunya) (Giessing, Hundepool and Castro (2009)). When applying the same scheme to the protection of animal production statistics of the European Union two unforeseen features of CTA were required: it should deal with non-additive tables, and it should cope with negative protection levels. While the former is a simple extension, the latter significantly changes the optimization model; even worse, the solution space of the models with negative protection levels increases (as shown in Section 4), and it may make harder finding an optimal or good solution. Non-additivity may result when dealing with externally obtained tables, with empty or approximate cells. Negative protection levels can be used to deal with correlated tables. More details will be provided at the beginning of sections 3 and 4. In this paper we discuss several models for the general CTA problem with either positive and negative protection levels, and either additive or non-additive tables. The computational results show which is the most effective variant to be used in practice for real-world instances. The most efficient model turned out to be as efficient as the standard CTA model, being much more general: it can deal with either additive or non-additive tables, and with positive and negative protection levels.

The structure of the paper is as follows. Section 2 outlines the standard CTA formulation, which is the basis for the extensions of subsequent sections. Sections 3

and 4 show the new models to deal with non-additive tables and negative protection levels. Section 5 reports the computational results obtained with the several resulting models in the solution of a set of real-world instances.

2. The standard CTA model

Any CTA instance, either with one table or a number of tables, can be represented by the following parameters:

- A set of cells $a_i, i \in \mathscr{N} = \{1,\ldots,n\}$, that satisfy some linear relations $Aa = b$ (a being the vector of a_i's), and a vector $w \in \mathbb{R}^n$ of positive weights for the deviations of cell values.
- A lower and upper bound for each cell $i \in \mathscr{N}$, respectively l_{x_i} and u_{x_i}, which are considered to be known by any attacker. If no previous knowledge is assumed for cell i, then $l_{x_i} = 0$ ($l_{x_i} = -\infty$ if $a \geq 0$ is not required) and $u_{x_i} = +\infty$ can be used.
- A set $\mathscr{S} = \{i_1, i_2, \ldots, i_s\} \subseteq \mathscr{N}$ of indices of confidential or sensitive cells.
- A lower and upper protection level for each confidential cell $i \in \mathscr{S}$, respectively lpl_i and upl_i, such that the released values satisfy either $x_i \geq a_i + upl_i$ or $x_i \leq a_i - lpl_i$.

CTA attempts to find the closest safe values $x_i, i \in \mathscr{N}$, according to some distance L, that makes the released table safe. This involves the solution of the following optimization problem:

$$\begin{array}{rl} \min\limits_{x} & ||x-a||_L \\ \text{subject to} & Ax = b \\ & l_x \leq x \leq u_x \\ & x_i \leq a_i - lpl_i \text{ or } x_i \geq a_i + upl_i \quad i \in \mathscr{S}. \end{array} \tag{1}$$

Problem (1) can also be formulated in terms of deviations from the current cell values. Defining $z = x - a$, $l_z = l_x - a \leq 0$, and $u_z = u_x - a \geq 0$, we obtain

$$\begin{array}{rl} \min\limits_{z} & ||z||_L \\ \text{subject to} & Az = 0 \\ & l_z \leq z \leq u_z \\ & z_i \leq -lpl_i \text{ or } z_i \geq upl_i \quad i \in \mathscr{S}. \end{array} \tag{2}$$

Using the L_1 distance weighted by w, and introducing variables $z^+, z^- \in \mathbb{R}^n$ so that $z = z^+ - z^-$ and $|z| = z^+ + z^-$, and binary variables $y \in \{0,1\}^s$ the final MILP model for CTA is:

$$\min_{z^+,z^-,y} \quad \sum_{i=1}^{n} w_i(z_i^+ + z_i^-) \tag{3a}$$

$$\text{subject to} \quad A(z^+ - z^-) = 0 \tag{3b}$$

$$0 \le z_i^+ \le u_{z_i}, \quad 0 \le z_i^- \le -l_{z_i} \;\; i \in \mathscr{N} \setminus \mathscr{S} \tag{3c}$$

$$y \in \{0,1\}^s \tag{3d}$$

$$\left.\begin{array}{ll} upl_i y_i & \le z_i^+ \le u_{z_i} y_i \\ lpl_i(1-y_i) & \le z_i^- \le -l_{z_i}(1-y_i) \end{array}\right\} i \in \mathscr{S} \tag{3e}$$

Constraints (3b) impose feasibility of the published perturbed table. Constraints (3c) guarantee perturbations are within allowed bounds. Constraints (3d)–(3e) force the new table to be safe. When $y_i = 1$ the constraints mean $upl_i \le z_i^+ \le u_{z_i}$ and $z_i^- = 0$, thus the protection sense is "upper"; when $y_i = 0$ we get $z_i^+ = 0$ and $lpl_i \le z_i^- \le -l_{z_i}$, thus the protection sense is "lower".

3. Non-additive tables

In some instances the original cell values do not satisfy $Aa = b$. This is mainly due to missing or approximate cell values of externally provided tables, which may require the application of cell imputation techniques. This is specially relevant for data managed by Eurostat, where the sources are different countries of the European Union. In particular, this requirement was necessary for the protection of animal production statistics (i.e., milk production) at the European Union and state members levels. Tables already protected (i.e., they contained missing information) for each member state were received. The protection of this set of tables, together with the European Union totals, can be accomplished by first estimating values for the missing information, although they result in non-additive tables, and using RCTA to make the resulting tables both safe and additive. Some details about the overall procedure can be found in Giessing, Hundepool and Castro (2009).

If the table is non-additive, i.e., $Aa \neq b$, then the constraints (3b) of the CTA model have to be replaced by

$$A(z^+ - z^-) = b - Aa. \tag{4}$$

Indeed, note that a deviation satisfying (4) makes the resulting table feasible:

$$A(a + z^+ - z^-) = Aa + A(z^+ - z^-) = Aa + (b - Aa) = b.$$

If the original table is already additive, then $b - Aa = 0$, and therefore (3b) and (4) are equivalent. Since (4) is more general, it should be preferred in any CTA model. Note the complexity of (3) is the same either considering (3b) or (4).

4. Negative protection levels

Negative protection levels may be required when protecting correlated tables. Protection levels lpl_i and upl_i for cell a_i preclude values of the interval $[a_i - lpl_i, a_i + upl_i]$ for this cell in the released table. Let us refer to this interval as the "protection interval". If the protection levels are positive then $a_i \in [a_i - lpl_i, a_i + upl_i]$, which is the usual situation. However, if this table is correlated with another that has been previously protected and released, we may need a protection interval that does not include a_i (for instance, to avoid that the ratios between both released tables are close to their real values). Of course if a_i is not in the protection interval, it may be released with no change, and then it could be (wrongly) assumed it is no longer a confidential cell, and that it does not require protection levels. However, because of the deviations of other cells and the preservation of the constraints (4), a positive deviation of a_i may be required in a solution, in which case the protection interval has to be considered. This issue of negative protection levels is directly related with the non-additivity of the previous section. In particular, for the real case of the European Union animal production statistics project (i.e., milk production), the presence of non-additive tables (whose values were estimated) may mean that the protection intervals have to be shifted, which may result formally in negative protection levels. Additional details can be found in Giessing, Hundepool and Castro (2009).

According to the signs of the lower and upper protection levels, there are four possible combinations that should be addressed by the new CTA model. Note that the MILP model (3) used, for instance, in Castro (2006) and Dandekar and Cox (2002) is only valid for one case, when protection levels are nonnegative. On the other hand, the generic formulation (2) is valid for the four cases, but it is not in the form of a mathematical programming problem. For instance, for a cell $a_i = 10$ with lower and upper protection levels lpl_i and upl_i, the four cases according to signs imposed by the constraints of (2) are:

- If $lpl_i = 3$ and $upl_i = 2$, then $z_i \leq -3$ or $z_i \geq 2$, i.e., the protection interval is $[7, 12]$.
- If $lpl_i = 3$ and $upl_i = -2$, then $z_i \leq -3$ or $z_i \geq -2$, i.e., the protection interval is $[7, 8]$.
- If $lpl_i = -2$ and $upl_i = 3$, then $z_i \leq 2$ or $z_i \geq 3$, i.e., the protection interval is $[12, 13]$.
- If $lpl_i = -2$ and $upl_i = -3$, then $z_i \leq 2$ or $z_i \geq -3$, i.e., any value can be released for this cell (there is no protection interval).

If the constraints (3e) were applied when protection levels are negative, then some components of z^+ or z^- would be negative, and the objective function (3a) would no longer represent the absolute value. This happens because in (3e) variables z^+ and z^- are associated to upper and lower protection deviations, instead of being auxiliary variables to model the L_1 distance.

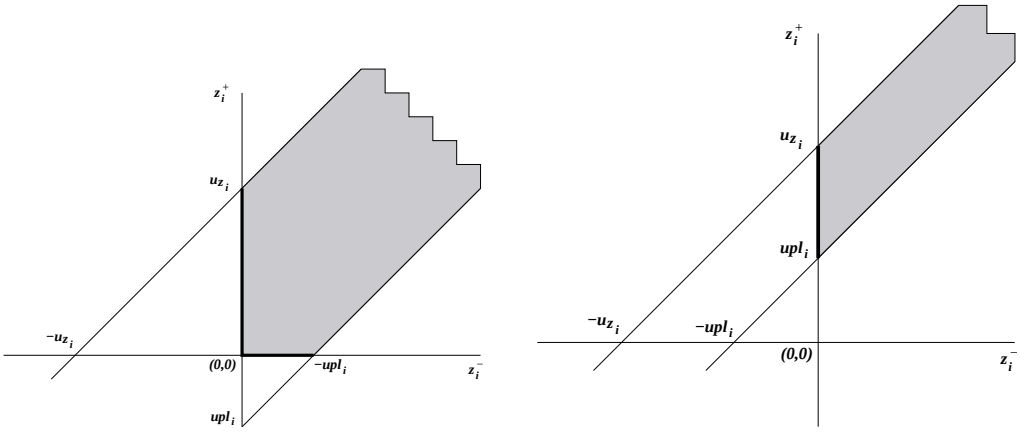

Figure 1: *In grey, feasible set Ω^i for $y_i = 1$, when either $upl_i \leq 0$ (figure (a)) or $upl_i \geq 0$ (figure (b)).*

Let us consider the model (2), and let us introduce $z^+, z^- \in \mathbb{R}^n$ such that $z = z^+ - z^-$ and $|z| = z^+ + z^-$. Then, considering the table may be non-additive, (2) can be written as

$$\begin{array}{ll} \min\limits_{z^+,z^-} & \sum\limits_{i=1}^{n} w_i(z_i^+ + z_i^-) \\ \text{subject to} & A(z^+ - z^-) = b - Aa \\ & l_z \leq z^+ - z^- \leq u_z \\ & z_i^+ - z_i^- \leq -lpl_i \text{ or } z_i^+ - z_i^- \geq upl_i \quad i \in \mathscr{S} \\ & (z^+, z^-) \geq 0. \end{array} \tag{5}$$

Introducing binary variables $y \in \{0,1\}^s$, (5) can be recast as the following MILP model:

$$\begin{array}{ll} \min\limits_{z^+,z^-,y} & \sum\limits_{i=1}^{n} w_i(z_i^+ + z_i^-) \\ \text{subject to} & (z^+, z^-, y) \in \Omega = \Omega^A \cap (\cap_{i\in\mathscr{N}} \Omega^{0_i}) \cap (\cap_{i\in\mathscr{S}} \Omega^i), \end{array} \tag{6}$$

where Ω^A, Ω^{0_i} and Ω^i are defined as

$$\Omega^A = \left\{(z^+, z^-) : A(z^+ - z^-) = b - Aa\right\}, \tag{7}$$

$$\Omega^{0_i} = \left\{(z_i^+, z_i^-) : l_{z_i} \leq z_i^+ - z_i^- \leq u_{z_i}, (z_i^+, z_i^-) \geq 0\right\} \; i \in \mathscr{N}, \tag{8}$$

$$\begin{aligned} \Omega^i = \big\{&(z_i^+, z_i^-, y_i) : z_i^+ - z_i^- \geq upl_i y_i + l_{z_i}(1 - y_i), \\ &z_i^+ - z_i^- \leq -lpl_i(1 - y_i) + u_{z_i} y_i, (z_i^+, z_i^-) \geq 0, y_i \in \{0,1\}\big\} \; i \in \mathscr{S}. \end{aligned} \tag{9}$$

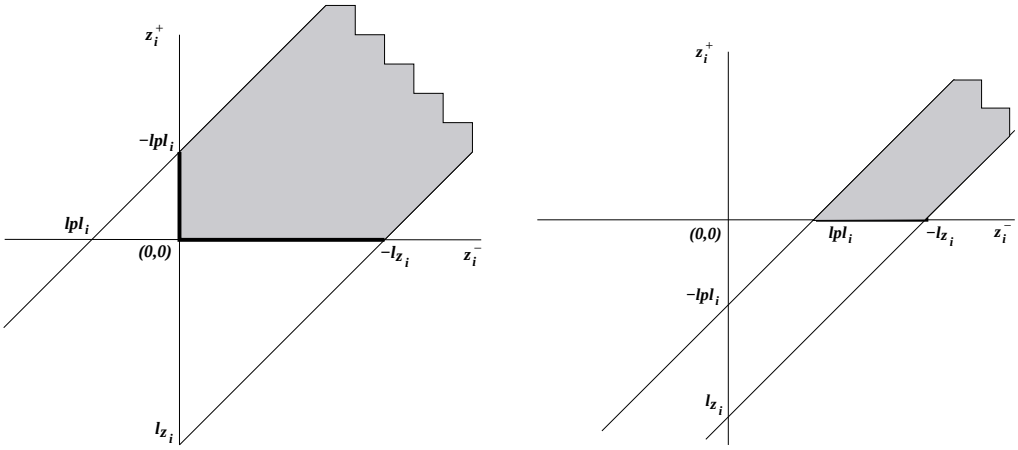

Figure 2: *In grey, feasible set Ω^i for $y_i = 0$, when either $lpl_i \leq 0$ (figure (a)) or $lpl_i \geq 0$ (figure* (b)*).*

If $y_i = 1$, Ω^i reduces to

$$\left\{(z_i^+, z_i^-) : upl_i \leq z_i^+ - z_i^- \leq u_{z_i}, (z_i^+, z_i^-) \geq 0\right\} \tag{10}$$

i.e., the protection sense is "upper". If $y_i = 0$, Ω^i is made up of points

$$\left\{(z_i^+, z_i^-) : l_{z_i} \leq z_i^+ - z_i^- \leq -lpl_i, (z_i^+, z_i^-) \geq 0\right\}, \tag{11}$$

i.e., the protection sense is "lower". (10) and (11) define the feasible sets on the (z_i^+, z_i^-) space for the deviations of sensitive cells, depending on they are, respectively, upper or lower protected. The feasible set (10) is shown in Figure 1, considering two different cases: either $upl_i \leq 0$ – Figure 1*(a)* – or $upl_i \geq 0$ – Figure 1*(b)*. Similarly, Figure 2 shows the feasible set (11) for the two cases $lpl_i \leq 0$ – Figure 2*(a)* – and $lpl_i \geq 0$ – Figure 2*(b)*. Note that when $lpl_i = 0$ and $upl_i = 0$ both figures *(a)* and *(b)* of Figures 1 and 2 coincide.

From the objective function of (6), since $w_i > 0$, we have that in an optimal solution either $z_i^+ > 0$ or $z_i^- > 0$, but not both. Therefore, the optimal sets of Figures 1 and 2 are restricted to the thick segments on the axes. When lpl_i and upl_i are nonnegative, once y_i is fixed, the optimal sets are convex and we know which component will be zero in the optimal solution: $z_i^- = 0$ if $y_i = 1$ (Figure 1*(b)*), and $z_i^+ = 0$ if $y_i = 0$ (Figure 2*(b)*). Therefore we may write an alternative formulation for Ω^i when $lpl_i \geq 0$ and $upl_i \geq 0$:

$$\begin{aligned}\Omega_1^i = \big\{&(z_i^+, z_i^-, y_i) : upl_i y_i \leq z_i^+ \leq u_{z_i} y_i, \\ &lpl_i(1-y_i) \leq z_i^- \leq -l_{z_i}(1-y_i), y_i \in \{0,1\}\big\} \ i \in \mathscr{S}.\end{aligned} \tag{12}$$

Note that constraints in Ω_1^i are equal to constraints (3e) of the standard CTA model. Next Proposition 1 shows that formulation (12) is stronger than (9). Moreover, denoting by $LR(\Omega)$ the linear relaxation of the set Ω (i.e., the set obtained by replacing conditions $y_i \in \{0,1\}$ in Ω by $0 \leq y_i \leq 1$ in $LR(\Omega)$), the proposition also shows that the linear relaxation of (12) is included in that of (9), and therefore any branch-and-bound based procedure is in theory more efficient with formulation Ω_1^i.

Proposition 1 *Given the two sets defined in (9) and (12), if* $lpl_i \geq 0$ *and* $upl_i \geq 0$, *then*

(i) $\Omega_1^i \subset \Omega^i$, *for all* $i \in \mathscr{S}$;

(ii) $LR(\Omega_1^i) \subset LR(\Omega^i)$, *for all* $i \in \mathscr{S}$.

Proof

(i) The proof is immediate just looking at Figures 1*(b)* and 2*(b)*.

(ii) We first show that $LR(\Omega_1^i) \subseteq LR(\Omega^i)$. From (12), any point (z_i^+, z_i^-, y_i) in $LR(\Omega_1^i)$ satisfies

$$\begin{array}{rcl} upl_i y_i \leq & z_i^+ & \leq u_{z_i} y_i, \\ l_{z_i}(1-y_i) \leq & -z_i^- & \leq -lpl_i(1-y_i). \end{array}$$

Adding the two previous inequalities we obtain $upl_i y_i + l_{z_i}(1-y_i) \leq z_i^+ - z_i^- \leq u_{z_i} y_i - lpl_i(1-y_i)$, and thus, from (9), $(z_i^+, z_i^-, y_i) \in LR(\Omega^i)$. Finally we show that $LR(\Omega_1^i) \neq LR(\Omega^i)$ by noting that, for instance, when $y_i = 0$ points in $LR(\Omega_1^i)$ are of the form $(0, z_i^-, 0)$ (i.e., the thick line of Figure 2*(b)*), while points in $LR(\Omega^i)$ are of the form $(z_i^+, z_i^-, 0)$ (i.e., the shadowed region of Figure 2*(b)*). □

Similarly, the thick lines of Figure 3 show the subset of Ω^{0_i} in an optimal solution. Such a subset is nonconvex, and it can be improved by adding two new groups of constraints:

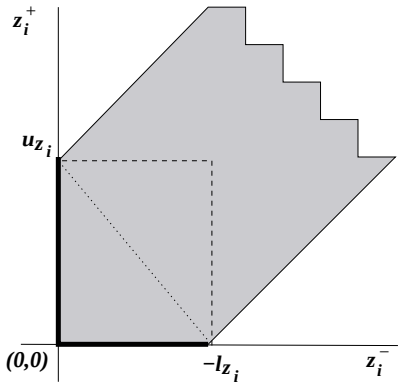

Figure 3: *Strengthened formulations for* Ω^{0_i}, *represented by the shadowed region. Additional constraints are shown by the dashed and dotted lines.*

- First, we may add upper bounds for z_i^+ and z_i^-. These are represented by the dashed line of Figure 3. The new set

$$\Omega_1^{0_i} = \left\{(z_i^+, z_i^-) : 0 \leq z_i^+ \leq u_{z_i}, 0 \leq z_i^- \leq -l_{z_i}\right\} \tag{13}$$

  is bounded, and there is no need to include the now redundant constraints $l_{z_i} \leq z_i^+ - z_i^- \leq u_{z_i}$, $i \in \mathscr{N}$. Note that (13) only imposes bounds on variables, but no constraint; this can significantly improve the performance of a solver.

- Second, looking at Figure 3 it is clear that the convex hull of points in the optimal set is the triangle of vertices $(0,0), (-l_{z_i}, 0), (0, u_{z_i})$. The convex hull is formulated by the new set

$$\Omega_2^{0_i} = \left\{(z_i^+, z_i^-) : z_i^+ \leq u_{z_i} + \frac{u_{z_i}}{l_{z_i}} z_i^-, (z_i^+, z_i^-) \geq 0.\right\} \tag{14}$$

  The new constraint $z_i^+ \leq u_{z_i} + \frac{u_{z_i}}{l_{z_i}} z_i^-$ corresponds to the dotted line of Figure 3. Although it reduces the feasible region, it complicates the formulation by adding an extra constraint for each cell $i \in \mathscr{N}$, which could significantly increase the computational time.

The following proposition 2 states the previous relations between sets Ω^{0i}, $\Omega_1^{0_i}$ and $\Omega_2^{0_i}$.

Proposition 2 *Given the sets Ω^{0i}, $\Omega_1^{0_i}$ and $\Omega_2^{0_i}$ respectively defined in (8), (13)and (14), then $\Omega_2^{0_i} \subset \Omega_1^{0_i} \subset \Omega^{0i}$.*

Proof The proof is immediate from Figure 3. □

4.1. Models considered

Combining the alternative formulations for Ω^i and Ω^{0_i} of previous section, (i.e., either Ω^i or Ω_1^i, and either Ω^{0i}, $\Omega_1^{0_i}$ or $\Omega_2^{0_i}$) in (6), it is possible to obtain different optimization models. We note that the alternative formulation Ω_1^i for Ω^i can only be used if $lpl_i \geq 0$ and $upl_i \geq 0$, whereas the alternative formulations $\Omega_1^{0_i}$ and $\Omega_2^{0_i}$ for Ω^{0_i} are always valid.

We have considered eight different models, which are tested in the computational results of Section 5. The objective function is the same for the eight models, and corresponds to that of (6); the models only differ in the representation of the feasible set. The first group of four models considers the formulation Ω^i for any $i \in \mathscr{S}$, independently of the sign of lpl_i and upl_i (i.e., even when $lpl_i \geq 0$ and $upl_i \geq 0$ formulation Ω^i is used). These four models will be denoted as the *new* models, and their feasible sets are respectively formulated as:

$$\Omega_{new_1} = \Omega^A \cap (\cap_{i\in\mathscr{N}} \Omega_1^{0_i}) \cap (\cap_{i\in\mathscr{S}} \Omega^i), \quad (15)$$

$$\Omega_{new_2} = \Omega^A \cap \left(\cap_{i\in\mathscr{N}} (\Omega_1^{0_i} \cap \Omega^{0_i})\right) \cap (\cap_{i\in\mathscr{S}} \Omega^i), \quad (16)$$

$$\Omega_{new_3} = \Omega^A \cap \left(\cap_{i\in\mathscr{N}} (\Omega_1^{0_i} \cap \Omega_2^{0_i})\right) \cap (\cap_{i\in\mathscr{S}} \Omega^i), \quad (17)$$

$$\Omega_{new_4} = \Omega^A \cap \left(\cap_{i\in\mathscr{N}} (\Omega^{0_i} \cap \Omega_1^{0_i} \cap \Omega_2^{0_i})\right) \cap (\cap_{i\in\mathscr{S}} \Omega^i). \quad (18)$$

The second group of four models uses Ω^i for sensitive cells $i \in \mathscr{S}$ with either $upl_i < 0$ or $lpl_i < 0$, and Ω_1^i when $upl_i \geq 0$ and $lpl_i \geq 0$. They are thus a hybrid between the standard CTA model of (3) and the general model for negative protection levels of (6). They will be referred as the *hybrid* models. Making a partition of the set of sensitives cells $\mathscr{S} = \mathscr{S}^- \cup \mathscr{S}^+$, where $\mathscr{S}^- = \{i \in \mathscr{S} : lpl_i < 0 \text{ or } upl_i < 0\}$ and $\mathscr{S}^+ = \{i \in \mathscr{S} : lpl_i \geq 0 \text{ and } upl_i \geq 0\}$, the feasible sets of the four hybrid models are:

$$\Omega_{hyb_1} = \Omega^A \cap (\cap_{i\in\mathscr{N}} \Omega_1^{0_i}) \cap (\cap_{i\in\mathscr{S}^+} \Omega_1^i) \cap (\cap_{i\in\mathscr{S}^-} \Omega^i), \quad (19)$$

$$\Omega_{hyb_2} = \Omega^A \cap \left(\cap_{i\in\mathscr{N}} (\Omega_1^{0_i} \cap \Omega^{0_i})\right) \cap (\cap_{i\in\mathscr{S}^+} \Omega_1^i) \cap (\cap_{i\in\mathscr{S}^-} \Omega^i), \quad (20)$$

$$\Omega_{hyb_3} = \Omega^A \cap \left(\cap_{i\in\mathscr{N}} (\Omega_1^{0_i} \cap \Omega_2^{0_i})\right) \cap (\cap_{i\in\mathscr{S}^+} \Omega_1^i) \cap (\cap_{i\in\mathscr{S}^-} \Omega^i), \quad (21)$$

$$\Omega_{hyb_4} = \Omega^A \cap \left(\cap_{i\in\mathscr{N}} (\Omega^{0_i} \cap \Omega_1^{0_i} \cap \Omega_2^{0_i})\right) \cap (\cap_{i\in\mathscr{S}^+} \Omega_1^i) \cap (\cap_{i\in\mathscr{S}^-} \Omega^i). \quad (22)$$

5. Computational results

The eight optimization models resulting from the feasible sets defined by (15)–(22) and the objective function of (6) have been implemented using the AMPL modelling system (Fourer, Gay and Kernighan (2002)). A set of real-world instances have been solved with the eight models, using both the MILP solvers of CPLEX 12.1 and Xpress Optimizer 19.00.00. The particular values of w in these real-world instances were specifically computed (i.e., they were neither 1, nor the cell value). All the runs have been performed on a Linux Dell Precision T5400 workstation with 16GB of memory and four Intel Xeon E5440 2.83 GHz processors, without exploitation of parallelism capabilities (to fairly compare CPLEX and Xpress solution times, since our CPLEX version allows multithreading whereas the Xpress version do not). A MILP optimality gap of 0 was set for all the executions. The MILP optimality gap is defined as

$$gap = \frac{|best - lb|}{1 + |best|} \cdot 100\%, \quad (23)$$

best being the best current solution, and *lb* the best current lower bound. A zero optimality gap is impractical with real-world instances as the ones considered in this

work, since it provides prohibitively large executions. However, it was used to test the strength of each formulation.

Feasibility and integrality tolerances were also reduced for both solvers; they were set, respectively, to 10^{-8} and 0 for CPLEX and to 10^{-8} and 10^{-8} for Xpress (since it does not allow integrality tolerances smaller than the feasibility tolerance). Such a reduction is required to avoid solutions with underprotected cells. Indeed, (9) and (12) impose, among other constraints,

$$z_i^+ - z_i^- \leq -lpl_i(1-y_i) + u_{z_i} y_i, \qquad z_i^+ \leq u_{z_i}\, y_i.$$

In practical tables u_{z_i} and l_{z_i} may be very large, e.g., $u_{z_i} = l_{z_i} = M$. If, because of the feasibility and integrality tolerance, we get a solution $y_i = \epsilon$ instead of $y_i = 0$, then the above constraints would be

$$z_i^+ - z_i^- \leq -lpl_i(1-\epsilon) + M\epsilon \neq -lpl_i, \qquad z_i^+ \leq M\epsilon \neq 0.$$

Therefore, sensitive cell i would result underprotected. Decreasing the feasibility tolerance, we make the above ϵ value smaller, but the problem becomes much harder and the probability of the problem being reported as infeasible – when it is feasible – is increased. A better option is to avoid big M values for cell deviations, but this means the real cell bounds (lower and upper bounds) should be small. In this work we set a bound $M = 10^8$ for cell deviations (i.e., if the real bound is greater than M, then it is replaced by M; otherwise the real bound is used). However, even with such a bound on the deviations and with the above small feasibility and integrality tolerances, some solutions reported unprotected cells, as shown in below tables.

Table 2 shows the dimensions of the real-world instances considered, which were generated in Statistics Germany from data provided by Eurostat. Columns n, s, m and "N.coef" report, respectively, the number of cells, sensitive cells, linear relations of the table, and nonzero coefficients of matrix A. The nine instances can be grouped in small instances (the first three), medium size instances (the middle three), and large instances (the last three). The medium and large instances can be considered difficult since they

Table 2: *Dimensions of the test instances.*

Instance	n	s	m	N.coef
APS-Jan	87	5	35	177
APS-Feb	87	5	35	177
APS-Mar	87	5	35	177
sbs-E	1430	382	991	4680
sbs-C	4212	1135	2580	13806
dposrel	9568	1492	3956	22698
sbs-D_a	28288	7142	13360	87022
sbs-D_b	28288	7131	13360	87022
balofpay-eus-p1	39060	2483	37818	175965

Table 3: *Results for each model and solver (three smaller instances).*

Instance	CPLEX				Xpress			
model	CPU	f^*	B&B	n.u.	CPU	f^*	B&B	n.u.
APS-Jan								
new_1	0.004	7.12	0	0	0	7.12	1	0
new_2	0.004	7.12	0	0	0	7.12	1	0
new_3	0.004	7.12	0	0	0	7.12	1	0
new_4	0	7.12	0	0	0	7.12	1	0
hyb_1	0	7.12	0	0	0	7.12	1	0
hyb_2	0.004	7.12	0	0	0	7.12	1	0
hyb_3	0.004	7.12	0	0	0	7.12	1	0
hyb_4	0.004	7.12	0	0	0	7.12	1	0
APS-Feb								
new_1	0.008	66.85	6	0	0	66.85	15	0
new_2	0.008	66.85	6	0	0	66.85	15	0
new_3	0.004	66.85	11	0	0	66.85	15	0
new_4	0.004	66.85	11	0	0	66.85	15	0
hyb_1	0.008	66.85	6	0	0	66.85	3	0
hyb_2	0.004	66.85	6	0	0	66.85	3	0
hyb_3	0.004	66.85	7	0	0	66.85	3	0
hyb_4	0.004	66.85	7	0	0	66.85	3	0
APS-Mar								
new_1	0.008	11.90	1	0	0	11.90	1	0
new_2	0.004	11.90	1	0	0	11.90	1	0
new_3	0.004	11.90	3	0	0	11.90	1	0
new_4	0.004	11.90	3	0	0	11.90	1	0
hyb_1	0.004	11.90	0	0	0	11.90	1	0
hyb_2	0.004	11.90	0	0	0	11.90	1	0
hyb_3	0.004	11.90	0	0	0	11.90	1	0
hyb_4	0	11.90	0	0	0	11.90	1	0

have a complex structure, and a significant number of cells, constraints and sensitive cells. These nine instances are related to data from structural business statistics, balance of payment, and animal production statistics of the European Union.

The results for each model and solver, for each group of three instances, i.e., small, medium size and large, are respectively reported in Tables 3–5. Columns "CPU", f^*, "B&B" and "n.u." provide, respectively, the CPU solution time, best objective function reached, number of branch-and-bound nodes explored, and number of underprotected cells in the solution. A time limit of 7200 seconds was set in all the executions. When this time limit is reached, the CPU time column shows the optimality gap (23) of the solution obtained within the time limit. We provide results with both CPLEX and Xpress since they are the two solvers mainly used in the statistical disclosure control community.

Table 4: *Results for each model and solver (three medium size instances).*

Instance	CPLEX				Xpress			
model	CPU	f^*	B&B	n.u.	CPU	f^*	B&B	n.u.
sbs-E								
new_1	42.86	107442.27	7406	0		(1)		
new_2		(1)				(1)		
new_3	364.71	107720.37	107090	0		(1)		
new_4	167.49	107439.65	37401	0		(1)		
hyb_1	14.26	107442.27	1056	0		(1)		
hyb_2	12.73	107439.65	1086	0		(1)		
hyb_3	10.36	107853.98	770	0	(2)(1.05%)	121084.67	3014871	0
hyb_4	9.48	107853.26	885	0		(1)		
sbs-C								
new_1	(2)(0.07%)	313562.69	305971	0		(1)		
new_2	(2)(0.07%)	313655.95	213097	0		(1)		
new_3	(2)(46%)	314547.38	161825	0		(1)		
new_4	(2)(1.3%)	313742.96	192901	0		(1)		
hyb_1	58.70	331425.16	525	0		(1)		
hyb_2	52.69	315160.90	518	0		(1)		
hyb_3	904.37	324572.49	103510	0		(1)		
hyb_4	(2)(0.004%)	314001.24	1301687	0		(1)		
dposrel								
new_1	10.2	7807.98	1533	62	8	7808.28	961	0
new_2	9.9	7807.98	1422	62	10	7808.28	915	0
new_3	18.0	7807.98	1723	62	8	7813.72	517	0
new_4	18.8	7807.99	1943	63	8	7813.72	517	0
hyb_1	8.9	7808.28	1231	1	6	7808.28	299	0
hyb_2	8.5	7808.28	1238	1	5	7808.29	361	0
hyb_3	13.6	7808.28	1939	1	6	7813.72	311	1
hyb_4	13.7	7808.28	2047	1	6	7813.72	311	1

(1) No feasible solution found, problem reported as infeasible

(2) Time limit reached

However, our purpose is not to compare the two different solvers, but the models and to show the difficulties found by the optimization solvers. From Tables 3–5 the following observations can be made:

- Both CPLEX and Xpress, with the eight different models, successfully solved the very small instances of Table 3 in less than 1 second, exploring very few branch-and-bound nodes.
- The medium size and large instances of Tables 4–5 are difficult for state-of-the-art solvers. For some instances and models, CPLEX and Xpress were not able to

find either an optimal solution (executions marked with a [2] in Tables 4–5), or a feasible solution within the 7200 seconds time limit (executions marked with a [4] in Table 5). In some CPLEX executions the optimization process even failed by numerical errors of the solver (runs marked with a [3] in Table 5).

- For some combinations instance–model the optimization problems are reported as infeasible (when they are feasible) due to the small feasibility tolerances used. These executions are marked with a [1] in Tables 4–5. However, if the feasibility

Table 5: *Results for each model and solver (three larger instances).*

Instance	CPLEX				Xpress			
model	CPU	f^*	B&B nodes	n.u.	CPU	f^*	B&B nodes	n.u.
sbs-D$_a$								
new$_1$	[2](20%)	414666.45	26096	0		[1]		
new$_2$	[2](22%)	417332.53	20699	0		[1]		
new$_3$		[3]				[1]		
new$_4$	[2](33%)	417841.08	22207	0		[1]		
hyb$_1$		[1]				[1]		
hyb$_2$		[1]				[1]		
hyb$_3$		[4]				[1]		
hyb$_4$		[4]				[1]		
sbs-D$_b$								
new$_1$	[2](22%)	408432.48	29318	0		[1]		
new$_2$	[2](56%)	767929.98	16906	0		[1]		
new$_3$	[2](31%)	416436.74	19107	0		[1]		
new$_4$		[3]				[1]		
hyb$_1$		[4]				[1]		
hyb$_2$		[1]				[1]		
hyb$_3$		[4]				[1]		
hyb$_4$		[4]				[1]		
balofpay-eus-p1								
new$_1$		[3]			[2](88%)	5366.63	6407	0
new$_2$		[3]			[2](88%)	5366.63	6507	0
new$_3$		[3]			[2](88%)	7300.04	5351	0
new$_4$		[3]			[2](88%)	7300.04	5281	0
hyb$_1$		[3]			[2](54%)	4708.11	5727	0
hyb$_2$		[3]			[2](56%)	4554.76	9690	0
hyb$_3$		[3]			[2](55%)	5303.8	1672	0
hyb$_4$		[3]				[1]		

[1] No feasible solution found, problem reported as infeasible

[2] Time limit reached

[3] Unrecoverable failure: singular basis

[4] Time limit reached with no integer solution

tolerance is increased, then we obtain bad solutions, with a significant number of underprotected cells. This undesirable effect due to large feasibility tolerances even happens for small instances; for instance, four out of the five sensitive cells of Table 3 would have been underprotected in the optimal solution if a feasibility tolerance of 10^{-5} had been used. Even with the tight feasibility tolerances considered, we see that executions of instance "dposrel" of Table 4 provided 63 underprotected cells for the *new* models; this value was reduced to one underprotected cell when the *hybrid* model was used.

- The additional constraints (14) in models new_3, new_4, hyb_3 and hyb_4 may significantly increase the solution time. For instance, model new_1 of instance "sbs-E" with CPLEX takes 42.86 seconds, while models new_3 and new_4 take 364.71 and 167.49 seconds; similarly, for CPLEX and instance "dposrel", models new_3 and new_4, and hyb_3 and hyb_4, require a 100% and a 50% more time than models new_1 and new_2, and hyb_1 and hyb_2, respectively. However, as suggested by Proposition 2 the number of branch-and-bound nodes may be reduced: this is observed in *new* models of instance "dposrel" with Xpress, and *hybrid* models of instance "sbs-E" with CPLEX, both of Table 4. Therefore, constraints (14) could be of help in some situations.
- In general, the *hybrid* model is preferred, since it is more efficient. This is consistent with Proposition 1. For instance, in Table 4 for "sbs-E" and CPLEX, the four executions with the *hybrid* models are much faster than with the *new* variants. This is also observed in instance "balofpay-eus-p1" and Xpress, where the *hybrid* models provided better solutions than the *new* models within the time limit. However, in some cases, when the *hybrid* models have difficulties, the *new* ones can be an alternative, as shown for instance sbs-D_a and CPLEX in Table 5.

6. Conclusions

From the computational and theoretical results with the several models tested, it can be concluded that the *hybrid* approach is in general more efficient than the *new* models for the solution of CTA instances with either positive or negative protection levels. It has also been shown that both types of models might have difficulties when exposed to real-world and complex CTA instances, even using the best today optimization solvers. This motivates further development on optimization methods for difficult CTA instances. Some steps have been done along these lines using, e.g., cutting plane or Benders decomposition approaches (Castro and Baena (2008)), and heuristic block coordinate decompositions (González and Castro (2009)). However, there is not yet a definitive approach for any CTA instance. This is part of the further work to be done in the statistical disclosure control field.

7. Acknowledgments

This work has been supported by grants MICINN MTM2009-08747 and SGR-2009-1122, and by Eurostat framework contract 22100.2006.002-226.532.

References

Castro, J. (2005). Quadratic interior-point methods in statistical disclosure control. *Computational Management Science*, 2(2), 107–121.

Castro, J. (2006). Minimum-distance controlled perturbation methods for large-scale tabular data protection. *European Journal of Operational Research*, 171, 39–52.

Castro, J. (2007a). A shortest-paths heuristic for statistical data protection in positive tables. *INFORMS Journal on Computing*, 19(4), 520–533.

Castro, J. (2007b). An interior-point approach for primal block-angular problems. *Computational Optimization and Applications*, 36, 195–219.

Castro, J. and Baena, D. (2008). Using a mathematical programming modeling language for optimal CTA. *Lecture Notes in Computer Science*, 5262, 1–12.

Castro, J. and Giessing, S. (2006). Testing variants of minimum distance controlled tabular adjustment. In *Monographs of Official Statistics. Work session on Statistical Data Confidentiality*, Eurostat-Office for Official Publications of the European Communities, Luxembourg, 2006, 333–343. ISBN 92-79-01108-1.

Castro, J., González, J. A. and Baena, D. (2009). User's and programmer's manual of the RCTA package. Technical Report DR 2009/01, Dept. of Statistics and Operations Research, Universitat Politècnica de Catalunya.

Cox, L. H., Kelly, J. P. and Patil, R. (2004). Balancing quality and confidentiality for multivariate tabular data. *Lecture Notes in Computer Science*, 3050, 87–98.

Dandekar, R. A., and Cox, L. H. (2002). Synthetic tabular data: an alternative to complementary cell suppression. Manuscript, Energy Information Administration, U.S. Department of Energy.

Domingo-Ferrer, J. and Franconi, L. (eds.) (2006). *Lecture Notes in Computer Science. Privacy in Statistical Databases*, 4302, Springer, Berlin.

Domingo-Ferrer, J. and Saigin, Y. (eds.) (2008). *Lecture Notes in Computer Science. Privacy in Statistical Databases*, 5262, Springer, Berlin.

Fair Isaac Corporation Dash Xpress (2008). *Xpress Optimizer. Reference Manual, release 19.0.0*.

Fourer, R., Gay, D. M. and Kernighan, D. W. (2002). *AMPL: A Modeling Language for Mathematical Programming*, Duxbury Press.

Giessing, S., Hundepool, A. and Castro, J. (2009). Rounding methods for protecting EU-aggregates. In *Worksession on statistical data confidentiality. Eurostat methodologies and working papers*, Eurostat-Office for Official Publications of the European Communities, Luxembourg, 255–264.

González, J. A. and Castro, J. (2011). A heuristic block coordinate descent approach for controlled tabular adjustment. *Computers & Operations Research*, 38, 1826–1835.

IBM ILOG CPLEX (2009), *User's Manual for CPLEX v12.1*.

Salazar-González, J. J. (2006). Controlled rounding and cell perturbation: statistical disclosure limitation methods for tabular data. *Mathematical Programming*, 105, 583–603.

Willenborg, L. and de Waal, T. (2000). *Lecture Notes in Statistics. Elements of Statistical Disclosure Control*, 155, Springer, New York.