A General Model for the Answer-Perturbation Techniques

Daniel Stamate , Henri Luchian [*]
Faculty of Computer Science, "Al.I.Cuza" University of Iasi,
Romania

Ben Paechter
Napier University of Edinburgh,
UK

Abstract

Answer-perturbation techniques for the protection of statistical databases were introduced in [7]; they are flexible (perturbation kept under control), modular (do not interact with the DBMS) techniques, which compare favorably to previous protection techniques. In this paper, we generalise the answer-perturbation techniques w.r.t. the operation used for modifying the exact answers (thus enhancing the level of protection). Experimental results are also included; they indicate statistical soundness of our techniques.

1. Introduction

Approximate answers to user queries has shown lately to be of interest even in the frame of usual, nonstatistical databases (for certain real-time applications); in scientific and statistical databases, when users are performing exploratory analysis [9], approximate answers for queries may be sufficient [5]. Therefore it is natural that statistically-controlled perturbation models form an important topic in the study of statistical databases (SDB). Various such models were proposed using either simple operations (e.g., the addition / multiplication of/by a value of a random variable -r.v.- to/of the actual values in the database) or complicated ones (e.g., generating a dummy database which preserves the distribution of the values in the original database); see [2], [10], [6] and for an excellent survey, [1].

We have presented a model of the same kind (*answer-perturbation techniques* - [8]), which has the important property of not interacting with the Database Management System (modularity): the final result of the query evaluation provided by the DBMS is perturbed - outside the DBMS - by means of values of r.v.'s and this perturbed answer is provided to the user; the Database Administrator has complete control on the level of perturbation.

No proof exists that one specific operation would be "best" (in terms of protection and usability - see [8]) to be used for perturbing the data (or the query sets, or the exact answers - depending on the protection technique) within a specific model. In this paper we generalise the actual operation being used for perturbing the exact answers; we study the usability and the protection in this frame. As in our previous paper, all the random variables used for perturbing the exact answers have the mean close to 1 (which keeps a low level of "noise" added to the original information by means of perturbation).We also present some experimental results which indicate that the answer-perturbation techniques preserve the original distribution of the exact answers to queries and that the new operations proposed in this paper increase the level of protection, while keeping the distribution of the provided answers close to the original one.

2. Overview of the answer-perturbation techniques

The answer-perturbation approach is sketched in fig. 1[8]

[*] Henri Luchian (hluchian@uaic.ro) and Daniel Stamate (stamate@uaic.ro) are joint first authors of this paper.

0-8186-6610-2/94 $04.00 © 1994 IEEE

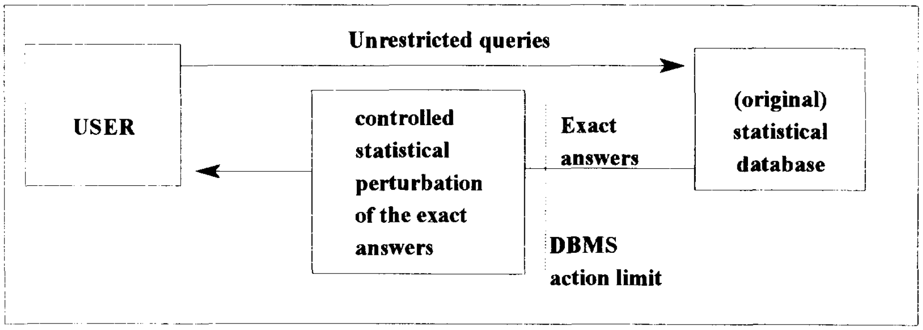
Unrestricted queries
USER
controlled statistical perturbation of the exact answers
Exact answers
(original) statistical database
DBMS action limit

Fig.1. The answer-perturbation approach

To illustrate the way these techniques work, let **q** be a query and **r** the value of the exact answer to it. The user receives the answer **r*y**, where **y** is a number around **1** - see fig. 2 [8]. In the security models described in [8], we considered only the multiplication as the operation '*'.

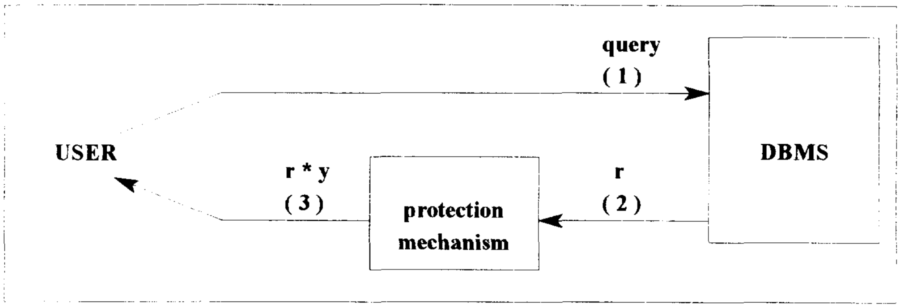

Fig. 2.

In [8] we have defined two notions: *usability* and *protection*. The usability **U** of a statistically protected SDB, **U**∈**[0,1]**, measures how near to one another are the exact answer to a query and the answer provided to the user (the user's viewpoint). The protection **P**, **P**∈**[0,1]**, shows the degree of security (in the worst case) of the database *(the DBA viewpoint)*; it is closely related to *statistical-disclosure control* ([4]). **U=0** shows that the answers provided to the users are unpredictably far from the exact answers, while **U=1** indicates that exact answers and provided answers coincide. On the other hand, **P=1** would indicate the highest level of protection, while **P=0** shows that a user could deduce (almost for sure) exact answers.

Of the four models we have presented in [8], we give here the general lines of the "delayed" model. Let $\{Y_1 ,...,Y_p\}$ be a set of r.v.'s and ν be a r.v. which takes each value from $\{1,2,...,p\}$ with probability p_i, **i=1..p**; $Y_1 ,...,Y_p$, ν are considered to be globally stochastically independent and Y_i, **i=1..p** have the mean $E(Y_i)$ close to **1**.

A r.v. Y_i, once chosen (ν(ω)=i), will be used for modifying the answers to m consecutive queries (see fig. 3 [8]), where **m**∈**N** is a parameter chosen by the DBA. If the same query **b** is asked many times, the answer is successively provided as: $r \cdot y_1 , r \cdot y_2 ,..., r \cdot y_m , r \cdot y_{m+1} ,..., r \cdot y_{2m} , r \cdot y_{2m+1},..., r \cdot y_{3m} ,...$ where **r** is the exact answer, $y_1 ,...,y_m$ are values of the r.v. Y_{i_1}; $y_{m+1} ,...,y_{2m}$ are values of the r.v. Y_{i_2}, etc. ; $i_1 ,i_2 ,...$ are values of the r.v. ν. We have, in this model, a local usability (for the answers between positions $(k-1) \cdot m+1$ and $k . m$ - for any fixed **k** - where a single r.v. Y_{i_k} is in use) and a global usability (for answers perturbed by at least two r.v.'s).

The local usability is:

$$U_{i_k} = g(\mid E(Y_{i_k} - 1 \mid , \mathrm{Var}(Y_{i_k})) \qquad (1)$$

where **g** is a continuous function, **g:[0,∞)X[0,∞)→[0,1]**, strictly decreasing in each of the two arguments, with **g(0,0)=1** (see [8]) and **Var(X)** is the variance of **X**.

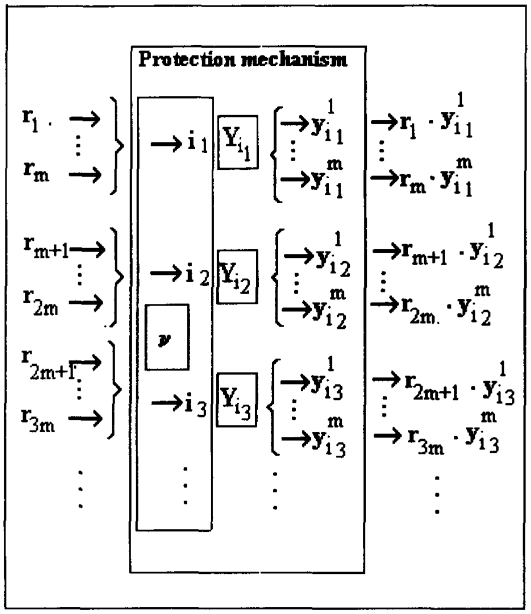

Fig.3. The 'delayed' model.

Our first model used only one r.v., **Y**, for perturbing the exact answers. Since in answer-perturbation techniques all queries are allowed at any time, an intruder can improve the level of usability granted by the Database Administrator, by repeatedly asking the same query. Theoretically, if a query **q**, for which the exact answer is **r**, is repeated indefinitely, then the value **r · E(Y)** can be computed (in practice: approximately) by taking the arithmetic mean **A** of all provided answers. If **r** is known from other sources, then **E(Y)** can be computed from the equation **r · E(Y) = A** and hence all the exact answers can be evaluated. In the delayed model, the level of protection was improved. Nevertheless, if users are allowed to repeat indefinitely the same query, the above mentioned drawback still exists: from a theoretical point of view, in the "delayed" model it only takes longer for an intruder to reach the same results as in the first model.

3. A general model

In the general setting we introduce here, the perturbation has two components:

3.1. Deterministic perturbation related to the value r.

The provided answer should have the form **r·φ(r)**, where φ takes values in a neighborhood of **1** (for usability reasons): **φ:R→[1-a,1+a]**. In order to compromise the database, an intruder has to know in advance many exact answers, $\mathbf{r_1}, \mathbf{r_2}, \ldots, \mathbf{r_n}$; then he can deduce $\varphi(r_1),\varphi(r_2),\ldots,\varphi(r_n)$ and eventually make some inferences if φ has certain regularity properties (e.g., monotonicity, continuity, etc.).

Example. If φ is an increasing function, then so is **x·φ(x)**. Suppose an intruder knows in advance the exact answers $\mathbf{r_1}$ and $\mathbf{r_2}$ and gets the provided answers $r_1 \cdot \varphi(r_1)$ and $r_2 \cdot \varphi(r_2)$. Then, for any provided answer **r·φ(r)** which satisfies $r_1 \cdot \varphi(r_1) \leq r \cdot \varphi(r) \leq r_2 \cdot \varphi(r_2)$, the intruder will infer that $r_1 \leq r \leq r_2$; if $|r_1 - r_2|$ is very small, the intruder obtains a good approximation of **r**.

This example shows that φ should be chosen to be a non-monotonic, non-continuous function.

3.2. Random perturbation.

As in our previous models, randomness will also appear in the provided answers. Namely, the users will get answers of the form **r·φ(r)·x**, where **x** is a value of a r.v. **X** with codomain **[1-ε,1+ε]**.One can consider that $\mathbf{X_r} = \varphi(\mathbf{r}) \cdot \mathbf{x}$ is a r.v. corresponding to the exact answer **r**. Randomness is beneficial in increasing the level of protection (while preserving reasonable bounds for usability and keeping it under control). Indeed, under random perturbation, an intruder has to ask the same query indefinitely in order to obtain only one value of the form **r·φ(r)·E(X).**

With the two components of the perturbation in mind, the usability can be defined as:

$$\mathbf{U=u(t_1 ,t_2)} \qquad (2),$$

where $\mathbf{t_1}$ *is the mean value of* $\mathbf{T_1(r)=|1-\varphi(r) \cdot E(X)|}$, $\mathbf{t_2}$ is the mean value of $\mathbf{T_2(r)=\varphi^2(r) \cdot Var(X)}$; $\mathbf{u:R_+^2 \to [0,1]}$ is a decreasing function in each of the two arguments (note that **u(0,0)=1**).

Further extensions of the model can be obtained by considering n functions, $\varphi_1, \varphi_2, \ldots, \varphi_n$, which the DBA interchanges.

4. φ Functions

We give four examples of functions φ, chosen from the large class of non-monotonic, non-continuous functions for which the codomain has the form $[\mathbf{1-a, 1+a}]$.

a) Let $\mathbf{a=0.1}$; we construct a function $\varphi_1:\mathbf{R}\rightarrow[\mathbf{0.9,1.1}]$. Let $\mathbf{r}\in\mathbf{R}$, $\mathbf{r>0}$ and let

$$\mathbf{y} = \{\text{if } \mathbf{r}\in\mathbf{N} \textit{ then } \mathbf{1/r} \textit{ else } \mathbf{r} \textit{ endif.}\}$$

Let $\mathbf{b_1,\ldots,b_{10}}$ be the first ten digits after the decimal point in the binary representation of $\mathbf{y}$ and let $\mathbf{z=0.b_1\ldots b_{10}}$ $(\mathbf{0\le z<1})$. We define $\varphi_1(\mathbf{r})=(\mathbf{2\cdot z-1})\cdot\mathbf{a+1}$ $(\varphi_1(\mathbf{r})\in[\mathbf{1-a,1+a}))$.For $\mathbf{r<0}$, define $\varphi_1(\mathbf{r})=\varphi_1(|\mathbf{r}|)$ and for $\mathbf{r=0}$, define $\varphi_1(\mathbf{r})=\mathbf{1}$.

b) φ_2 is defined similar to φ_1, with $\mathbf{1/r}$ replaced by $\mathbf{sqrt(r)}$.

c) Let $\mathbf{d}\in\mathbf{R}_+$. Define: φ_3 (r) =

$\{$ *if* ($[\mathbf{3r/d}]$ is even) *then* φ_1 (r) *else* φ_2 (r) *endif* $\}$,

$\mathbf{d}$ is the step (for the argument $\mathbf{r}$) used for interchanging φ_1 and φ_2.

d) φ_4 $(\mathbf{r})=(\mathbf{z-0.5})\cdot\mathbf{0.1+1}$, where $\mathbf{z=0.z_3z_4z_5z_6}$, $\mathbf{z_3}$, $\mathbf{z_4}$, $\mathbf{z_5}$, $\mathbf{z_6}$ are the 3rd, 4th etc. digits in the decimal representation of $\mathbf{sqrt(|r|+1)}$ (or $\mathbf{sqrt(|r|+2)}$, if $\mathbf{sqrt(|r|+1)}$ is an integer).

For $\mathbf{i=1..4}$ we have: $\varphi_i(\mathbf{x})\in[\mathbf{0.9,1.1}]$, $\forall$ $\mathbf{x}\in\mathbf{R}$; φ_i is non-monotonic.

5. A theoretical approach

5.1. Usability concerns.

Let $\mathbf{r}$ be the exact answer to a query $\mathbf{q}$ and $\mathbf{x}$ a value of the r.v. $\mathbf{X}$. Let "*" be a binary operation, which will describe the method used by the DBA for perturbing the exact answers (the provided answer will be $\mathbf{r*x}$). "*" should be known only by the DBA. In the previous section, $\mathbf{r*x} = \mathbf{r}\cdot\varphi(\mathbf{r})\cdot\mathbf{x}$. Since one can think that the r.v. which is actually used for perturbing the exact answer $\mathbf{r}$ is $\mathbf{r*X}$, the arithmetic mean of many provided answers in response to query $\mathbf{q}$ will approximate $\mathbf{E(r*X)}$. Therefore, for usability reasons, $\mathbf{E(r*X)}$ should be a good approximation of $\mathbf{r}$. Hence we have the following condition:

$$\max_{\mathbf{r}} \mathbf{f_1(r)\le b_1}, \qquad (3)$$

where $\mathbf{f_1(r)=|(E(r*X)-r)/r|}$ and $\mathbf{b_1}$ is very small. A similar argument for variance gives a second condition:

$$\max_{\mathbf{r}} \mathbf{f_2(r)\le b_2}, \qquad (4)$$

where $\mathbf{f_2(r)=Var(r*X)/r^2}$ and $\mathbf{b_2}$ is also very small. As it is easily seen, the smaller $\mathbf{b_1}$ and $\mathbf{b_2}$, the higher the usability.

5.2. Protection concerns.

A good level of protection would require:

$$\mathbf{f_1(r)\neq0,\ f_2(r)\neq0} \qquad (5)$$

(in practice, for any convenient norm, $||\mathbf{f_1}||$ and $||\mathbf{f_2}||$ should not be very small). If we consider the function $\mathbf{h_x(z)=z*x}$, then another condition to be satisfied in order to achieve a good protection level is that $\mathbf{h_x(z)}$ should not be continuous and / or monotonic (this will minimize the set of possible inferences).

5.3. The general model.

In order to avoid the explicit use of $\mathbf{r}$ (in $\mathbf{f_1}$ and $\mathbf{f_2}$), we will replace the means given by $\mathbf{t_1}$ and $\mathbf{t_2}$ in (2) by different expressions of means, $\mathbf{M_1}$ and $\mathbf{M_2}$, where:

$$\mathbf{M_i=(\textstyle\int_a^b f_i\,(r)dr)/(b-a),\ i=1,2} \qquad (6)$$

$(\mathbf{I=[a,b]}$ is the minimal interval which contains the values of all the exact answers; $\mathbf{f_1}$ and $\mathbf{f_2}$ are considered integrable). Usability can then be expressed as:

$$\mathbf{U=u(M_1\,,M_2)}, \qquad (7)$$

where $\mathbf{u}$ is the decreasing function from (2).

If an intruder tries to increase the usability by taking the mean of many answers provided in response to the same query, he gets:

$$(\mathbf{h_{x_1}(r)+\ldots+h_{x_n}(r)})\,/\,\mathbf{n}, \qquad (8)$$

where $\mathbf{x_1,\ldots,x_n}$ are values of the r.v. $\mathbf{X}$. According to the strong law of large numbers, the expression in (6) converges almost for sure to $\mathbf{E(r*X)}$; these approximations are also very close to $\mathbf{r}$. At this point, the operation "*" makes a difference: if "*" was multiplication, since $\mathbf{E(r\cdot X)=r\cdot E(X)}$, it is possible to separate $\mathbf{r}$ from the expression $\mathbf{E(r*X)}$.

An intruder who knows $\mathbf{r}$ can obtain valuable information about $\mathbf{X}$ (to be eventually used for computing other exact answers from similar arithmetic means). If, on the other hand, "*" is a more complicated operation, then the most favorable case for the intruder is when there exists an operation "$*_1$" such that $\mathbf{E(r*X)=r*_1E(X)}$; this would allow him to separate the exact answer from the random part. In our example, if $\mathbf{r*x=r\cdot\varphi(r)\cdot x}$, then:

$$\mathbf{E(r*X)= r\cdot\varphi(r)\cdot E(X)=r*_1E(X)} \quad (9)$$

For the intruder , "$*_1$" is not completely specified, since φ is not known. In the models we proposed in [8], the task of an intruder was much simpler, since there we have $\mathbf{\varphi\equiv1}$ $\mathbf{(r*x=r\cdot x)}$. "*" can be chosen in such a way that "$*_1$" does not even exist.

5.4. Previous knowledge of the intruder.

When studying the protection, one has to always consider the most favorable case for the intruder. Therefore, assume that an intruder knows in advance the exact answers $\mathbf{r_1, r_2,..., r_p}$ to $\mathbf{p}$ queries $\mathbf{q_1, q_2,..., q_p}$ and also $\mathbf{\varphi(r_1)}$ (and, obviously, the protection technique) .' Then, by repeating $\mathbf{m_i}$ times $\mathbf{(i=1..p)}$ each of the $\mathbf{p}$ queries, the intruder can obtain the equations:

$$\mathbf{r_i\cdot\varphi(r_i)\cdot E(X)=v_i,\ i=1..p} \quad (10)$$

where $\mathbf{v_i}$ is the arithmetic mean of the $\mathbf{m_i}$ answers provided in response to query $\mathbf{q_i}$. From the first equation, the intruder will compute $\mathbf{E(X)}$ and from the other equations and previous knowledge he will obtain $\mathbf{\varphi(r_1),...,\varphi(r_p)}$. Computing these values is however a useless task. Indeed, suppose that, using $\mathbf{\varphi(r_1),...,\varphi(r_p)}$, the intruder obtains a good interpolation of φ; then, each time he will need an exact answer $\mathbf{r}$, he will be able to obtain (an approximate value of) $\mathbf{r\cdot\varphi(r)}$ and he would need the value $\mathbf{r}$ in order to interpolate $\mathbf{\varphi(r)}$! If necessary, the DBA can make disclosure a more difficult task if the function φ and/or the r.v.$\mathbf{X}$ are changed from time to time.

5.5. Increasing the usability.

After querying $\mathbf{p}$ times the same query, an intruder can obtain a higher level of usability: $\mathbf{U_h=u(M_1,(1/p)\cdot M_2)}$. This is done by considering the arithmetic mean of the $\mathbf{p}$ answers as a single answer provided after perturbing the exact answer $\mathbf{r}$ using the r.v. $\mathbf{Y=(r*x_1+...+r*x_p)/p}$, with $\mathbf{E(Y)=E(r*X)}$ and $\mathbf{Var(Y)=(Var(r*X))/p}$. This leads to a level of protection (in the worst case) given by: $\mathbf{P=1-u(M_1,0)}$.

5.6. Complexity concerns.

In this general model, the complexity added to that of the "delayed" model ([8]) is due to computing φ. For the examples above $(\varphi_i,\ \mathbf{i=1..4})$, the extra-complexity is, in each case, within a constant.

5.7. Formulas for usability.

We now show that (7) is a general formula for usability: both the formula which defines the usability in [8] (similar to (1)) and the new formula (2) (for the case $\mathbf{r*x=r\cdot\varphi(r)\cdot x}$) can be derived from (7). Indeed, when "*" is "$\cdot$" (multiplication), $\mathbf{f_1}$ and $\mathbf{f_2}$ from (3) and (4) do not depend upon $\mathbf{r}$:

$$\mathbf{|(E(r*X)-r)/r|=|(r\cdot E(X)-r)/r|=|E(X)-1|},\text{ and}$$

$$\mathbf{Var(r*X)/r^2 =Var(r\cdot X)/ r^2 =Var(X)}.$$

Therefore, $\mathbf{M_1=|E(X)-1|}$ **and** $\mathbf{M_2=Var(X)}$, hence in this case (1) and (7) (considered for the same r.v. $\mathbf{X}$) coincide, with $\mathbf{g\equiv u}$ (for the definition of g, see (1) and the discussion in [8]). In other words, the formula for usability given in [8] is generalized by (7).

Furthermore, when $\mathbf{r*x=r\cdot\varphi(r)\cdot x}$, we get:

$$\mathbf{|(E(r*X)-r)/r|=|(r\cdot\varphi(r)\cdot E(X)-r)/r|=|\varphi(r)\cdot E(X)-1|}$$

and $\mathbf{Var(r*X)/r^2 =\varphi^2(r)\cdot Var(X)}$,

hence in this case (7) gives $\mathbf{U=u(M_1,M_2)=u(t_1,t_2)}$ and (7) also generalizes (2).

6. Experimental results

Since our models of protection are modular, i.e. do not interact with the DBMS, we do not have to consider values in the database, query sets, etc. Our experiments start where the DBMS action stops: we are concerned only with the transformation of exact answers to queries into perturbed answers. We have studied the perturbation from a statistical point of view: given the statistical distribution of the exact answers and the distribution of the r.v. used for perturbation, what will be the distribution of the perturbed answers?

We have considered both the model from [8], with perturbation made by means of multiplication (the "delayed" model in [8], with p=1 and the set of r.v.'s **{X}** - **V≡1**), and the general model presented above, with the same r.v. **X** and $\mathbf{r^*x = r \cdot \varphi_4(r) \cdot x}$. We considered the exact answers to have a normal distribution and then we studied - by means of the Kolmogorov-Smirnov test [3] - the distribution of the answers provided in each case. The r.v. **X** used for perturbation was **N(1,0.0125)**.

We have generated 15 sets of 1000 "exact answers" each, using the normal distribution with μ**=50** and σ**=10**. Column A of table 1 presents the first ten entries in one set. Columns B and C contain the corresponding perturbed answers by means of multiplication alone (B) and by means of $\mathbf{r^*x = r \cdot \varphi_4(r) \cdot x}$ (column C). We stress that the differences between the exact answers and the provided answers can be tuned by the DBA, using appropriate values for the parameters of the distribution of **X** or by choosing appropriate φ functions.

A	B	C
53.5830	54.5618	56.2144
60.2494	59.7143	60.4308
48.2024	47.3771	47.1118
60.5199	59.6526	58.7352
64.5181	65.2228	64.7819
42.9883	43.8224	42.6676
49.9883	49.5389	49.1847
52.3791	51.6423	52.2104
57.1734	57.3152	58.5463
52.2793	50.9859	53.1630
....	....	....

Table 1

For each of the 15 sets of "exact answers" we have computed the arithmetic mean (column A in table 2) and the variance (column B) of the perturbed answers. Then we have calculated the Kolmogorov-Smirnov statistics (column C in table 2):

$$\mathbf{KS=\max_{x \in R}|F_{1000}(x)-F(x)|}, \qquad (11)$$

where **F** is the theoretical repartition function (in our case, the repartition function corresponding to **N(50,10)**) and $\mathbf{F_{1000}}$ is the empirical repartition function of the 1000 perturbed answers. Small values of **KS** indicate that $\mathbf{F_{1000}}$ is very close to **F** and the hypothesis that the 1000 provided answers have a distribution almost identical to **N(50,10)** is accepted. For 1000 degrees of freedom and a 10% significance level, the value 0.038703 is the threshold under which this hypothesis is accepted.

The results are given in table 2. A pair of rows refers to one set of "exact answers": the first one corresponds to perturbation by r.v. **X** alone (the "delayed" model in [8], with **p=1**) and the second one corresponds to perturbation by both φ_4 and **X** (the general model).

No. of set	A	B	C
1.	49.2587	92.2384	0.0470
	49.1808	93.0170	0.0579
2.	49.9027	100.7112	0.0211
	49.9278	102.0551	0.0214
3	49.4241	103.5678	0.0348
	49.4748	107.2833	0.0413
4	50.3470	101.0460	0.0300
	50.3358	103.1129	0.0231
5	49.3065	95.2949	0.0420
	49.2587	98.8199	0.0454
6.	50.2609	102.5901	0.0214
	50.3328	105.0158	0.0236
7.	50.4919	101.5536	0.0273
	50.4442	103.0261	0.0241
8.	50.1040	95.7073	0.0190
	50.1733	97.9807	0.0139
9.	50.1257	102.5411	0.0268
	50.1577	104.5985	0.0141
10.	49.4393	103.1896	0.0270
	49.4086	104.3200	0.0288
11.	50.3613	98.7199	0.0314
	50.3662	99.6420	0.0272
12.	49.6917	100.3894	0.0199
	49.6968	102.2536	0.0232
13.	50.3327	95.3659	0.0292
	50.2941	97.3911	0.0248
14.	50.1615	106.9297	0.0293
	50.1536	107.9266	0.0238
15.	49.3201	95.1122	0.0379
	49.3556	96.7604	0.0355

Table 2

Note that the cases were **KS** > 0.038703 are due not only to the significance level we chose, but also to the fact that the original "exact answers" do not follow an ideal normal distribution. In ongoing experiments we consider different distributions for the exact answers and the r.v. **X**. We also intend to apply an adaptation of the Kolmogorov-Smirnov test which would allow us to directly compare the distributions of exact answers and provided answers, without using **N(**μ**,**σ**)**.

7. Conclusions

We have presented a generalized approach to the answer-perturbation techniques introduced in [8]. While preserving the modularity, efficiency and ability to control the perturbation of the original techniques, this generalization provides a higher level of protection, avoiding the (theoretical) possibility of compromising the statistical database by asking the same query indefinitely.

Experimental results show that the answer-perturbation techniques preserve the distribution of the exact answers.

References

[1]-Adam, N.R., Wortmann, J.C.: "Security-Control Methods for Statistical Data- bases:a Comparative Study", *ACM Computing Surveys*, vol.21, no.4, 1989, pp.515-556.

[2] - Beck, L.L.: "A Security Mechanism for Statistical Databases" -*ACM Transactions on Database Systems (TODS)*, vol.5, no.3, 1980, pp.316-338.

[3]-Ciucu, G., Craiu, V.:"Probability Theory and Statistics",**EDP**,Bucharest 1971 (in Romanian).

[4] - Dalenius, T. : "Towards a Methodology for Statistical Disclosure Control"- *Statistik Tidskrift*, vol.15, 1977, pp.429-444.

[5] - Hou, W.C., Ozsoyoglu, G.: " Statistical Estimators for Aggregate Relational Algebra Queries"- *ACM TODS*, vol.16, no.4, 1991, pp.600-654.

[6] - Liew, C.K., Choi, W.J, Liew,C.J.: "A Data Distorsion by Probability Distribution" - *ACM TODS*, vol.10, no.3, 1985, pp.395-411.

[7]-Luchian,H. ,Stamate,D. :"Statistical Protection for Statistical Databases ", Proc.of the 6th Int. Conf. SSDBMS, Ascona, Switzerland **ETH**, 1992, pp. 160-177.

[8]-Luchian, H., Stamate, D. : "Answer-Perturbation Techiques for the Protection of Statistical Databases", to apear in *Statistics and Computing*, **Chapman & Hall**.

[9]-Tukey,J.: "Exploratory Data Analysis", **Addison-Wesley**, Reading,Mass, 1977.

[10] Traub, J.F., Yemini, Y., Wozniakowski,H .:"The Statistical Security of a Statistical Database"-*ACM TODS*, vol.9, no.4, 1984, pp. 672-679.