Cell Selection for Fixed Interval Publication of Sensitive Cells to Protect Confidentiality

Bogong T. Li and Steve Cohen

Bureau of Labor Statistics
2 Massachusetts Ave. NE. Washington, D.C. 20212
li_t@bls.gov, and cohen_steve@bls.gov

Introduction

Bureau of Labor Statistics (BLS) collects labor market and other economic data from monthly surveys and censuses. On the one hand BLS obtains these data from business and individual respondents under a pledge of confidentiality, assumes the responsibility of protecting any individual respondent data or data summaries from being revealed; on the other hand, BLS makes maximum effort to deliver useful information to its users in the form of summarized tables. This results in a dilemma between protecting respondent confidentially and reducing information loss from withholding publication table contents.

BLS QCEW Proposed Publication Change

BLS Quarterly Census of Employment and Wages (QCEW) is a census that collects data under a cooperative program between BLS and the State Employment Security Agencies. The data contain broad employment and wage information for all U.S. workers covered by state unemployment insurance laws and federal workers covered by the Unemployment Compensation for Federal Employee program. QCEW serves as a census of monthly employment and quarterly wages information by 6-digit NAICS industry at the national, state, and county levels. Tabulations of QCEW outcomes are available by 6-digit NAICS industry, by county, by ownership sectors and by size groups, in the form of print, automatic e-mail, fax or plain text file directly from BLS Internet ftp servers (ftp://ftp.bls.gov/pub/special.requrest/cew). More information on BLS QCEW program is available on the QCEW website at http://www.bls.gov/cew/home.htm. The detailed coverage and readily availability of the QCEW tabular data make it especially vulnerable to confidentiality disclosure risks.

Many BLS survey programs currently use cell suppression for tabular data confidentiality protection. The cell suppression method completely suppresses the cells identified as having high confidentiality exposure risk as primary suppression cells based on certain confidentiality sensitivity measures, for example see Cox (1981). Additional cells, called secondary suppression cells, may also be suppressed, otherwise the values of the primary suppression cells can be calculated mathematically from published margins of the publication table. The cell suppression problem (CSP) involves finding a set of secondary suppression cells to protect against a potential attacker who aims to uncover contributing values from individual respondents to a cell of a statistical summary table. For example, QCEW tables publish employment levels and total wages of every NAICS industry at various detailed levels in U.S. by country, by size groups, by ownerships etc. An attacker could use mathematical techniques to calculate the contributing individual establishment employment levels or total wages based on published information contained in the table. A CSP solution would prevent the attacker from succeeding. CSP has been extensively studied in confidentiality protection literature, see for example Willenborg and de Waal (1996) for an in-depth discussion of this subject.

Since cell suppression methods currently implemented suppress a large number of cells in order to protect QCEW publication tables, an alternative method is sought. Using QCEW data analyzed in this paper, following the BLS confidentiality sensitivity measures, we found for this data set containing employment of five major industry sectors (2-digit NAICS sectors) within a medium-sized U.S. State, 9979 or 59% of 16,878 publication cells have to be completely suppressed using network method, 10631 or 62% of all cells using the hypercube method (for a description of the hypercube method see Repsilber (1994)). The level of employment represented by the suppressed cells is relatively small in comparison to the number of cells suppressed, ranging from 10% to 15% of the total value. Similar results of this magnitude for cell suppression have been also reported by other researchers. It is evident for the QCEW case that the cells suppressed in this manner comprise an

overwhelmingly large number of cells, while only represent a small portion of the total employment. Much detail on industry employment distribution at various geographic levels and other cross-classifications is lost due to confidentiality protection. Consequently users of BLS QCEW need as much information as possible on employment and wage information that are not sensitive but were blocked because of confidentiality. Users, including national policy makers, business firms, labor unions, trade associations, academic researchers and private research organizations hope BLS could reduce the number of suppressed cells or at least provide more information about these cells through some alternative method.

One alternative to complete suppression considered by QCEW would be to publish primary cells in pre-defined, fixed intervals (FIs). Instead of suppressing the value of a cell, this method would publish all primary suppression cells in pre-defined, fixed intervals (FIs) which cover the exact value of the sensitive cell value. The cut off points of these intervals are pre-defined and fixed. The consistency of the definition of these pre-defined intervals is kept across tables so that the users can compare values between various industries, geographic locations and other classifications by establishment characteristics. This method of publication can be used for employment, earnings and hours data, though our discussion in this paper will only focus on employment level data.

Similar to the issues surrounding CSP, if QCEW data is published with FIs replacing primary suppression cells, to prevent outside intruders gaining identifiable information of individual contributors to a cell, *additional* protecting cells (PCs) may have to be published in FIs. Otherwise an intruder may be able to utilize this additional information and the additive relationships existing in the table to estimate the value of primary cells now in FIs and therefore the value of some contributors to the cell. Intruders can produce better estimates now than before with the added information of published FI bounds. The problem of minimizing the amount of cell values now expressed in FIs by selecting the right set of PCs while still preserving the protection of primary cells is what we call the fixed interval publication problem (FIPP). FIPP therefore is the problem of finding additional PCs to protect the primary cells while minimizing the "information loss". If the concept of "information loss" can be consistently defined and measurable across the methods used to protect the tables -- just to mention it here without further deliberation, we found defining "information loss" consistently itself can be difficult which makes comparison between methods an imprecise game. This problem has its trace from the CSP therefore we will utilize as much as we knew about CSP to solve this problem. In this paper, we will propose a solution to the FIPP problem and evaluate the outcome from applying the solution to a subset of a realistic QCEW employment data set. We will use the following fixed interval ranges for employment levels: 0-19, 20-99, 100-299, 250-499, 500-999, 1000-2499, 2500-4999, 5000-9999, 10000-24999, 25000-49999, 50000-99999, 100000 or more. For example, QCEW would publish a suppressed cell value of "15" as "0-19", a value of "351" as "250-499", a value of "12,000" as "10,000-24,000".

Risks Associated with Publishing Primary Cells in FIs

Tables 1a-c show a comparison between publication tables of QCEW employment by 6-digit NAICS industry at the U.S. country level before and after the proposed change. Notice in the current publication table, Table 1b, cells that do not meet BLS confidentiality protection requirements and secondary protection cells are completely suppressed and are marked with "x".

Under the proposed publication changes, these suppressed cells will be published in pre-defined FIs, as seen in Table 1c. For example, a primary or secondary cell with employment value of "348" will be published as "250-499".

Table 1a. Fictitious Raw county employment table before any SDC treatment

NAICS code	*County 1*	*County 2*	*County 3*	*County 4*	*County 5*	*County 6*
623	604	328	2100	491	835	344
6231	280	138	650	377	357	301
62311	280	138	650	377	357	301
623110	280	138	650	377	357	301
6232	191	117	102	15	337	21
62321	168	-	102	8	251	-
623210	168	-	102	8	251	-
62322	23	117	-	7	86	21
623220	23	117	-	7	86	21
6233	133	71	1249	99	141	4
62331	133	71	1249	99	141	4
623311	2	54	895	-	118	-
623312	131	17	354	99	23	4

Table 1b. Current SDC treated publication table viewed by the public

NAICS code	*County 1*	*County 2*	*County 3*	*County 4*	*County 5*	*County 6*
623	604	328	2100	x	835	344
6231	x	x	650	x	x	301
62311	x	x	650	x	x	301
623110	x	x	650	x	x	301
6232	x	x	102	x	337	x
62321	168	-	102	x	251	-
623210	168	-	102	x	251	-
62322	x	x	-	7	86	x
623220	x	x	-	7	86	x
6233	x	x	1249	99	x	x
62331	x	x	1249	99	x	x
623311	x	x	895	-	x	-
623312	x	x	354	99	x	x

"x"s are cells not disclosable due to disclosure control.

Table 1c. Proposed new publication table with suppressed cells replaced by FIs

NAICS code	*County 1*	*County 2*	*County 3*	*County 4*	*County 5*	*County 6*
623	604	328	2100	250-499	835	344
6231	250-499	100-249	650	250-499	250-499	301
62311	250-499	100-249	650	250-499	250-499	301
623110	250-499	100-249	650	250-499	250-499	301
6232	100-249	100-249	102	0-19	337	20-99
62321	168	-	102	0-19	251	-
623210	168	-	102	0-19	251	-
62322	20-99	100-249	-	7	86	20-99
623220	20-99	100-249	-	7	86	20-99
6233	100-249	20-99	1249	99	100-249	0-19
62331	100-249	20-99	1249	99	100-249	0-19
623311	0-19	20-99	895	-	100-249	-
623312	100-249	0-19	354	99	20-99	0-19

This proposal publishes previously suppressed employment levels, regardless of whether a cell is a primary or secondary cell, in FIs, see Table 1c. From visualizing the table we quickly notice the benefit from this approach, (1) all cells are furnished with some numerical information (2) ordinal categories of the FI cells provide clear "level of magnitude" of cell values for comparison that would have been completely unavailable under suppression. More knowledge about employment is obtained from these consistent categorizations of suppressed cells. However, this approach also increases the risk of disclosing information from individual respondent to this survey. These risks include:

1. Attackers may obtain more precise estimates of primary cells through analyzing additive relationships existing in the table given additional information is provided. This is a serious problem. The attacker could estimate cell values through additive relationships in the table, i.e. the summed marginal of the rows and columns, and the sums among the hierarchical structures in a table through automated LP methods now available in computer programs. FI would make it easier for experienced users to combine their knowledge on an industry from other sources with these relationships to make estimates of individual reporters. To deal with this problem under the complete suppression assumption, such as the solutions to CSP, researchers developed sophisticate methods using linear programming (LP) and mixed integer linear programming (MILP) techniques to select additional cells to be completely suppressed, in a way such that all primary cells will be protected and the overall loss of information is minimized. However under the current proposal, by giving out the ranges or bounds of previously suppressed cells, attackers could improve their estimates of the primary cells that current CSP methods are to protect by incorporating additional information provided by FI bounds into their LP models. Additional secondary suppressions might have to be made.

2. Data users that are also contributors to publication cell may gain closer estimates of other contributors to the cell. As we know the precision of estimates to contributors of a cell made by other contributors to a cell defines the confidentiality sensitivity measure, i.e. higher the precision of the estimate, more sensitive the publication and vise versa. By knowing the published bound of primary cells, in this case, the FI bounds, a user and contributor to a cell may subtract its own value from either ends of the FI to have an estimate of another contributor of the cell. In the case where dominate contributor is present the estimate could come close, though the estimator can not be 100% sure without additional knowledge from other sources. Therefore the confidentiality sensitivity of the cell may significantly increase depending on the composition of contributors in the cell.

3. For non-single primary cells with the number of contributors below the usual threshold, for example, 2 or 3, published fixed interval bounds *sometimes* narrows the estimates of the dominate contributors made by other contributors within the cells. For example, in a cell with two contributors in which one is dominate, the smaller contributor is able to subtract its value from the published upper bound of FI and obtain a fairly close estimate of the dominate contributor.

4. For single count cells, one end of the fixed published range may be too close to the actual value of the single contributor. The contributor's value could even be the exact value of the boundary value of the FI. For example, a company with 18 employees could be the only 6-digit NAICS business within a county. The published value for this industry-by-county cell is 0-19. This company may or may not be comfortable with the published interval, even though the published employment level is in the form of a range, instead of an exact value.

We will focus on finding a solution to reduce risk 1 in this paper. The problem associated with risk 2 derives directly from the publication bounds of the primary cells and has nothing to do with any additive relationship existing in the table. Changing the bounds completely, or suppressing these primary cells completely is the only solution to the problem. However given all primary sensitive cells are published in FIs, no contributor to a primary cell can estimate a range of the value of other contributors with certainty. This additional doubt created by the boundaries adds a level of protection for all primary cells that were not available under complete suppression. Likewise, for risk 3, primary cells published in FIs would also provide additional ambiguities to the estimates made by any contributor of a cell or outside attackers to the value of other contributors. These ambiguities in the form of interval values, that would have been precise values under the CSP case, ensure any contributor is unable to *estimate* the "confidence interval" of other members with 100% precision. For risk 4, depending on the tolerance level of the sole contributor to a cell, we could choose to either suppress the count or the cell value to eliminate this risk.

To reduce risk 1, we start searching solutions made to solve CSP, since the risk 1 arises from the additive relationships in the table and is similar to CSP solution that have been implemented in some BLS survey programs. Our current knowledge

indicates CSP problem has been established by researchers as a MILP problem, see Kelly (1990). Exact solution to MILP model belong to the class of the strong *NP*-hard problem, (Kelly, Golden et al. 1992; Kao 1996), meaning that it is very unlikely an algorithm for the exact solution of CSP exists that guarantee a solution in polynomial-time for all practical statistical tables. The number of computations required for solving the CSP in MILP model grows exponentially with the size of the table. Therefore it will not be an option to protect large tables, as stated in Giessing (2001). For example, as pointed by Dula, Fagan, and Massell (2004), there are $(m \times n)(2k+1)$ number of variables for a $m \times n$ table with k primary cells. This is about 20 million variables for a 100×100 table with 1000 (10%) protected primary cells. This is beyond what today's fastest computer can carry out in a reasonable period of time. Other heuristic solution procedures such as the network flow method, see Cox (1980 and 1995), for 2-dimensional tables, multi-commodity network flow method for n-dimensional tables, see Castro and Nabona (1996) and hypercube method by Repsilber (1994) and Giessing (2001) have been proposed. These heuristic methods only provide sub-optimal solutions as pointed by Castro (2001). Fischetti and Salazar (1999) proposed a solution using branch-and-cut algorithm as one of the mathematical programming techniques to reach a solution with proven optimality on 2-dimensional tables with up to 500 rows and 500 columns. The problem is solved in a few minutes on a standard PC. Fischetti and Salazar-Gonzales (2000) extended their work to other tabular data including k-dimensional table with k>2, hierarchical tables, linked tables etc., using branch-and-cut based procedures. Alternatively, instead of completely suppressing table cells, Salazar (2001); Fischetti and Salazar (2003) proposed a "partial cell suppression" method that will publish a subset of table cells with variable estimation intervals. It has the advantage of having a polynomial-solvable solution that not only minimizes information loss but also saves the auditing phase after the selection. Though FIPP and CSP shares the same MILP model, *unfortunately*, so far we think all of the above mentioned secondary cell selection methods do not apply directly to selecting protecting cells (PCs) that are to be published in FIs, neither optimally nor heuristically. The reason is that these models can not accommodate the knowledge of the FI bounds. An optimal or near optimal selection method based on MILP models is yet to be found.

In this research we will propose an iterative "selection-improvement" algorithm, which improves cell selection upon each previous suppression pattern until all primary cells are sufficiently protected. All of the selection-improvement steps begin with procedures already implemented in BLS QCEW program. This approach is pragmatic since at the production level the program office does not need to completely re-program computers for a new linear programming based cell selection routine, except for adding some additional auditing and cell selection steps. Though no claim of optimality is made in this paper, this method does make publication of tables with FIs realistic, and, as the evaluation at the end shows, there aren't significantly more cells published as FIs than the number of cells completed suppressed. After describing our procedure, we will provide an evaluation study using actual employment data from a U.S. state. We will compare the results with current suppression methods, look into convergence rates, level of information loss and computer programming difficulties associated with various cell selection methods.

The Selection-Improvement Algorithm

The iterative selection-improvement algorithm has two stages at each iteration, (1) selecting PCs and (2) conducting an audit on the publication table with the newly selected PCs in FIs. If the audit finds any primary cell is still at risk, the algorithm re-iterates by selecting more PCs and conducting another audit until all primary cells are protected. The initial set of PCs is the set of cells selected through one of the CSP methods. In case the iterations fail at the end, i.e. no candidate PCs available for selection while there are still unprotected cells, the method defaults back to the usual CSP solutions targeting only the remaining exposed cells. The steps of the algorithm are summarized as follows:

Step 1. Identify primary and secondary cells in a table via a CSP method and publish them in pre-defined FIs.
Step 2. Apply linear constrained optimization to identify those primary cells with disclosure risks.
Step 3. For those primary cells at risk, select additional cells that have not been selected previously from the publication table and publish them in FIs. Three specific methods are proposed for this research and will be briefly described in following paragraph and sections. This is the 'selection step".
Step 4. Apply linear constrained optimization again to check if any primary cell in the original table is still at risk. If yes, return to step 3; otherwise EXIT the algorithm, the table is successfully protected. This is the "audit step".
Step 5. If the step 2 – 4 iteration fails to protect every primary cells, i.e. no further unsuppressed cells available for selection while there are still disclosed primary cells, use any solution method to CSP, i.e. completely suppress these exposure primary and corresponding secondary cells.

The following diagram illustrates the algorithm.

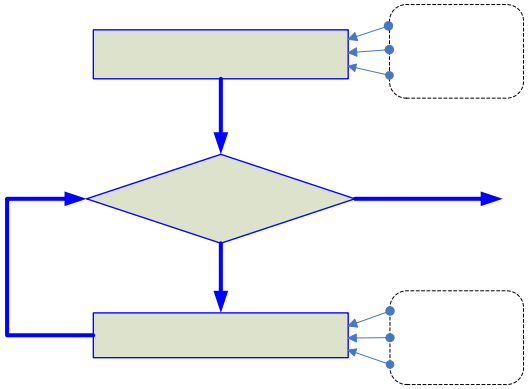

Figure 1. Proposed FIPP Solution

There are several alternative methods can be used to select additional PCs in Step 3. We can randomly select cells that are within the same row or column of the exposed primary cells, or we can select through more complex MILP models and mathematical programming techniques. We would like to minimize either the number of cells to be selected or the total value of the selected cells. In this paper we studied the following three methods in the selection step: the Systematic, Single-Source Shortest Path (SSSP) and the Random Selection methods. The methods are briefly described below and in more detail in the next section:

1. Systematic Method. To minimize values published in intervals, this method selects the smallest cell among all cells that form additive relationship with two selected exposure cells that need further protection that has not been suppressed during the previous iteration(s). This cell is published as a pre-defined FI. Default to Random Selection Method (see 3 next) at the end if this method fails.

2. Single-Source Shortest Path (SSPS) Method. This method models the table as a network similar to Traveling Salesman's Problem (TSP), treat all primary exposure cells on a table as destinations of a traveling map. The method aims to find the shortest path through these destinations, to minimize the total cell values expressed in FIs. To make this TSP solvable for all tables, the method fixes the order of the destinations or vertices on the table network. The method only needs to find the shortest path connecting the order-fixed set of vertices to form a closed "loop" with minimized path. Publish all cells that are not already selected in previous iterations on the chosen loop in FIs. Default to Random Selection Method if this method fails at the end.

3. Random Selection Method. This method randomly selects a cell among all cells that form additive relationship with the primary exposure cells. The candidate cells are cells that are either in the same row or column as the primary cell. If all cells forming additive relationships are already selected during previous iteration(s), or it by itself is the only decent from the higher hierarchy, go one hierarchy step higher until additional protecting cells can be found through additive relationships. Randomly select protecting cells among the candidates, publish these and all cells along the hierarchical searching path as FIs.

In addition to providing a valid solution, the FIPP algorithm introduced here is easy to implement in production, since it simply combines separate existing confidentiality protection procedures, such as the complementary cell suppression techniques and auditing of tabular data through linear programming. It requires less software changes in the survey production environment because the only change to current complementary cell selection procedures is the addition of auditing cycles. The difficulty of selecting additional PCs could be simple, for the Random Selection Method, or modestly complex, for the SSSP. The auditing of a table during any stage of the process can be done through available table auditing

software tools. Programming work for selecting PCs is only need to be done once and be reused later. More importantly, this method does not alter the actual micro data behind the tabular publication, as that of methods like adding noise to the micro data, which may add unwelcome noise to even safe cells.

Methods to Selection Additional PCs at the Selection Step

This section provides more detail and some justifications for the additional PC selection methods at the "selection step". There are other possibilities than what we are describing here, some of them are better than the others; we choose the following three to discuss as they represent different levels of difficulty. Once the selection step is done, the "audit step" of identifies publication cells which have estimated intervals narrower than the required protection range according to our chosen confidentiality protection rule. We denote the set of cells with exposure risk by their indices as $E = \{i_1, \cdots, i_p\}$, where i's are cell indices in a publication table. We need to select additional PCs among cells that are not published previously in intervals, and publish them in intervals in the next iteration(s). We denote in the k^{th} iteration the additional subset cells of E that are still at risk as E_k, k=0, 1, 2, …, K, $K \in N$, such that $E \supseteq E_0 \supseteq E_1 \supseteq E_2 \ldots \supseteq E_K$. We say a publication table is "safe" if and only if $E_K = \mathrm{O}$ and K is finite. Notice we restrict our selection of additional set of PCs targeting only E_k at step k, i.e. within previous risk cell subsets which are subsets of E_k, though audit at k^{th} iteration may indicate an exposure risk cell(s) that are outside of E_k. This is possible because secondary selection method does not guarantee protection of primary cells if secondary cells are published in intervals. Now that more cells are published in the form of FIs, cells that previously had no exposure risk now have exposure risk, since more information are available to the underlying LP model. We take this approach to make monitoring the iteration process easier by counting only those cells contained in E. We also denote the set of PCs selected at k^{th} iteration as F_k, k=0, 1, 2, …, K, $K \in N$. F_k s are mutually exclusive given the set indices are different. F_0 is an empty set, F_1 is the first PC set etc., F_K is the last set of PCs selected when the $E_K = \mathrm{O}$ condition is met etc. $\bigcup F_k$ is the final set of all PCs selected for the entire table.

Systematic Method. To select additional PC to protect all exposure cells found in the last iteration,, Systematic method begins by randomly selecting a pair of cells in E_k, say (i_p, i_q), where $p \neq q$, then identify all cells i such that they form additive relationships with both i_q and i_q. The cell with the smallest cell value among these cells is to be selected and put into F_k – nevertheless, we can alternatively select the cell with the smallest number of establishments contributing to the cell. In this way one additional PC is selected for every pair of exposure cells in E_k. If this is not possible, for example in the case all candidate PC are not available, we need to resort to Random selection method which we will discuss immediately next. In case the number of cells in E_k is odd, i.e. there will be one cell does not find a pair within E_k for it, choose the smallest cell that forms an additive relationship with that cell.. We then continue the selection step with an audit step. The audit step checks again if E_k is sufficiently protected by F_k. The cells in E_k which were not sufficiently protected are left to the set E_{k+1}, which requires additional PCs during the next iteration. Notice this process will continue if and only if $E_k \neq 0$, otherwise the table is declared "safe". During the $k+1$ iteration, Systematic method repeats itself and select PCs, followed by another audit step. Once $E_K = \mathrm{O}$ condition is met, we then publish the entire table with primary cells and cells in $\bigcup F_k$ in FIs. It is possible however $E_k \neq \mathrm{O}$ but $F_k = \mathrm{O}$, i.e. no further PC is available for selection while the table is still not fully protected. This happens because all available PCs have already been selected in previous iterations. In order to carry the process to completion, Systematic method uses the Random Selection method (to be discussed later). Random Selection method will select additional PCs in a way that will guarantee a full protection at the end of all iterations.

SSSP Method. This method differs from Systematic method only in how to select the additional PC set F_k. The iterative progress and stopping rule stay the same. This method utilizes an algorithm similar to the single-source shortest path (SSSP) algorithm, as explained in Huo (2004, p.308) that finds the shortest path between two vertices on a network connected by weighted edges. We view the publication table in the form of closed networks, where table margins form the vertices and the cells the connecting edges, for more discussion on statistical table's network conversion see Cox (1995). Cells in E_k are ordered on the network in a fixed fashion. The ends of the cells in E_k -- the cells are edges in the connected network, form the set of vertices we desire to find the shortest paths to connect them. The path of edges that along this shortest path will form the cells in F_k, the PC set in iteration k. The order of the vertices are fixed to avoid turning the problem to a Traveling Salesman's Problem that in theory is NP-hard, see Chartrand (1977), though by doing this we are not guaranteed to find the shortest path running through all vertices in E_k. During the selection step in iteration, we use an optimized algorithm to search through all paths joining the vertices formed by cells in E_k and select the path with the smallest cell value combined. These cells then will be selected to form F_k. The algorithm we use is similar to Dijkstra's algorithm for SSSP problem, see for example that described in Goodaire and Parmenter (1998), which labels the shortest path adjacent to all labeled vertices until the destination is also labeled (so called the "labeling algorithm"). Once the selection finishes, an audit step will proceed to check if any of E_k is still at exposure risk. Next iteration will proceed if and only if $E_K = \mathrm{O}$. It is possible $E_k \neq 0$ but $F_k = 0$, as explained earlier, i.e. no further PC is available for selection while there still are exposure primary cells. To complete the process, SSSP method uses the Random Selection method (to be discussed next). Random Selection method will guarantee a full protection at the end of the iterations.

Random Selection Method. Random Selection method differs from the previous two methods only in how it selects the additional PC set F_k. Systematic method minimizes the number of cells in F_k in each iteration, while the SSSP method minimizes total cell value in F_k. Both the Systematic and SSSP method do not guarantee a solution, because there could be no more table cells available for selection before all cells in E are completely protected. The Random Selection method is primarily designed to complete their processes, though it could be used alone from start. Random Selection method selects one additional PC randomly among all cells that are not previously selected and also form additive relationships with one cell in E_k that needs protection. Usually this cell in E_k is the cell fails the previous two methods. The auditing step will indicate if the table is "safe" at the end of step k, i.e. whether $E_K = \mathrm{O}$, otherwise the selection-audit iterative process continues. When all cells forming additive relationship with cell in E_k are already selected before E_k is protected, this is very likely for cells in E_k that are the only decedent in a hierarchical structure, Random Selection method chooses to step one or more steps up in the hierarchy to find PCs where are available, treating cells along the path as additional primary risky cells put into E_k. Random Selection is performed targeting these new primary risky cells as well as the original exposure cells in E_k. In practice we found in many instances that a cell with one hierarchy higher than an exposure cell is also an exposure cell therefore we have to use Random Selection method quite often in order to carry the whole process to finish.

Evaluation of the Method Using a Subset of Actual QCEW Data

We used actual QCEW employment publication tables for evaluating our stated FIPP procedure. This subset of QCEW data contains eight major 2-digit NAICS industry sectors in a medium-sized U.S. State. Table 2 displays the distribution of these industries and their establishment composition, i.e. the number of establishments and employment levels in each industry sector among these 2-digit NAICS industries in this data set. In actual BLS publications, these data are published in tabular forms separately in multi-dimensional table formats classified by county and 6-digit NAICS industry, as well as by establishment size group, metropolitan statistical area (MSA) and ownership types. We used only the 2-dimensional employment table classified by county and hierarchical NAICS code, from 2 to 6-digit, to demonstrate our algorithm. Uses of

2-diminensional table may limit our evaluation conclusion, since multi-dimensional publication tables are "connected", or in other words there are more additive relationships existing than what we considered. Nevertheless these additional additive relationships are identifiable. Once they are defined, we can always incorporate them in the model. Therefore we believe with some modification our method applies to tables with any dimensions and we should expect the number of cells in FIs somewhat more than we report here. Table 3 displays a portion of this publication table that a user may have seen in BLS publications. In this table the cells marked with "x" are suppressed cells due to primary and secondary suppressions. In this evaluation, we will apply our FIPP procedure to the data and compare their performance with that of the complete suppression.

Table 2. Study industry establishment population distribution

NAICS code	Industry	# of Establishments	Employment
31-33 (34)	Manufacturing	3,847	158,398
44-45	Retail Trade	15,563	288,980
48-49	Transportation and Warehousing	3,004	61,016
51	Information	704	16,805
52	Finance and Insurance	6,942	138,400
53	Real Estate and Rental and Leasing	4,504	45,839
54	Professional- Scientific- and Technical-Services	14,955	191,343
62	Healthcare and Social Assistance	11,326	265,607
	Total	*60,845*	*1,166,388*

It is obvious from the computing point of view that our solution poses some computing complications given the lack of available software, in the steps to produce table summary, formation of LP models and finding optimized solutions for the models etc. We will briefly describe here the computing procedure we took. The whole process is an integration of various computing tools that can carry the process to conclusion, though somewhat cumbersome. This summary below may be helpful for researchers interested in applying our method.

Software Tools Used to Select PCs and Audit Table

Modeling the linear additive relationships of the test publication table in the forms of a constrained linear optimization problem can be done through some mathematical programming languages, such as AMPL, MPS, GAMS etc.. Many commercial or free LP software are available to solve such models, Langohr and Gomez (2005) contains a good summary of solvers that implement these modeling formats including a web-based free solver at the Argonne National Laboratory, the NEOS (http://www-neos.mcs.anl.gov/). We used an approach for our problem that integrates free LP routine library lp_solve, see Berkelaar and Dirks et al. (2005) with S-plus® and Matlab® where lp_solve served as the Mixed Integer Programming (MIP) solver. lp_solve is a free (albeit GNU lesser general public license), open source MIP solver, with user manuals and FAQ information is available for downloading at http://groups.yahoo.com/group/lp_solve/ . It is a software programming routine library, called the API that can be called from almost any programming language to solve MIP problems, among them are C, C++, Pascal, Delphi, Java, VB, C#, VB.NET, Excel. There are two ways to pass the data to the library, via the API or input files. Standard lp_solve supports several input files types such as the commonly known MPS format, but it is not very readable. The format we used is the lp format that is more readable. We also found it is easier to use Matlab® as a medium when a publication has to be converted to lp format. We used a driver program called lp_solve that is callable from Matlab® via an external interface or MEX-function. As such, it looks like lp_solve is fully integrated with Matlab®. The complete interface of the driver is written in C so it has maximum performance.

For a quick note on how we process the data through these computer tools: we first put the raw micro data through primary and secondary suppression selection using software tool Tau-Argus, see Hundepool, Willenborg et al. (2004). The suppressed

table is then formatted to lp format in S-plus® to be used in Matlab®. In Matlab® we called solver lp_solve to conduct the audit of the table. If the iteration is not finished, the audited table is passed back to S-plus® and again we select additional PCs with the three methods we stated earlier in this paper. S-plus® and Matlab® were used to convert the publication table between publication table and LP model input formats. Additional PCs are selected within S-plus® where additive relationship of the entire publication table is kept. Unless the cycles successfully protected all primaries, the cycle should reiterate itself continuously. The convergence is guaranteed through the Random selection method. The software tools are not automatically connected to carry out the entire process so we have to manually transfer files between them. This is fortunately not difficulty since the number of iterations took only between 2 and 5, as we expected. The processing speed and virtual memory of S-plus® limits the size of the data we can process, which is another reason we decide to start with a single state data. This integration and task flow accomplished by each software tool is illustrated in the flow chart in Figure 2.

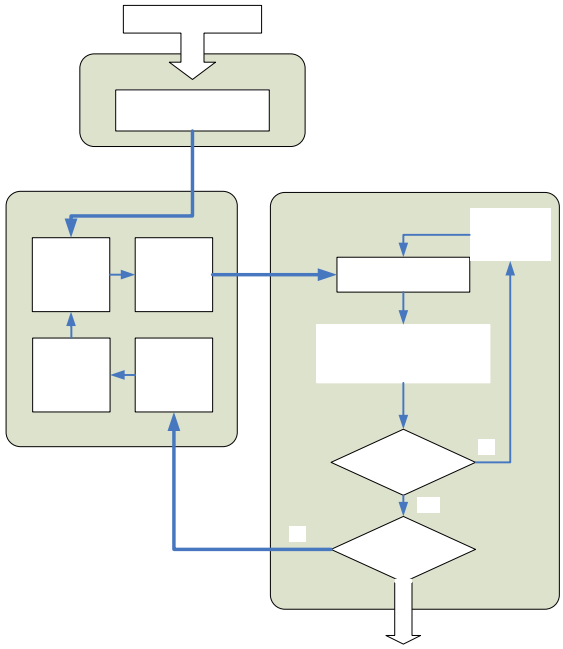

Figure 2. Integration of programming tools for the selection-improvement cycle

Table 3a. A sample section of the evaluation data set as it is currently published—slightly perturbed for confidentiality reasons

NAICS code	Counties of a U.S. State								
	Total	County 1	County 2	County 3	County 4	County 5	County 6	County 7	etc.
…	…	…	…	…	…	…	…	…	
…	…	…	…	…	…	…	…	…	
451	13940	113	1758	2691	111	x	241	64	
4511	9070	82	1121	1699	x	x	166	x	
45111	4187	26	703	773	89	-	51	51	
451110	4187	26	703	773	89	-	51	51	
45112	2648	x	274	451	x	x	x	-	
451120	2648	x	274	451	x	x	x	-	
45113	1237	x	110	302	-	x	x	x	
451130	1237	x	110	302	-	x	x	x	
45114	998	x	35	173	-	-	38	x	
451140	998	x	35	173	-	-	38	x	
4512	4870	31	637	992	x	-	75	x	
45121	3415	x	504	444	x	-	x	x	
451211	3193	x	x	438	x	-	x	x	
451212	222	x	x	6	-	-	-	x	
45122	1455	x	133	548	x	-	x	-	
451220	1455	x	133	548	x	-	x	-	
…	…	…	…	…	…	…	…	…	
…	…	…	…	…	…	…	…	…	
Total	1166388	15589	98129	190226	7524	5018	22485	12171	etc.

"x" are nondisclosable data due to primary and secondary suppressions

Table 3b. The same section of the evaluation data set as it is published under FIPP method

NAICS code	Counties of a U.S. State								
	Total	County 1	County 2	County 3	County 4	County 5	County 6	County 7	etc.
…	…	…	…	…	…	…	…	…	
…	…	…	…	…	…	…	…	…	
451	13940	113	1758	2691	111	0-19	241	64	
4511	9070	82	1121	1699	20-99	0-19	166	20-99	
45111	4187	26	703	773	89	-	51	20-99	
451110	4187	26	703	773	89	-	51	20-99	
45112	2648	0-19	274	250-499	0-19	0-19	0-19	-	
451120	2648	0-19	274	250-499	0-19	0-19	0-19	-	
45113	1237	0-19	110	302	-	0-19	20-99	0-19	
451130	1237	0-19	110	302	-	0-19	20-99	0-19	
45114	998	20-99	20-99	173	-	-	38	0-19	
451140	998	20-99	20-99	173	-	-	38	0-19	
4512	4870	31	637	992	0-19	-	75	0-19	
45121	3415	20-99	504	444	0-19	-	20-99	0-19	
451211	3193	20-99	250-499	438	0-19	-	20-99	0-19	
451212	222	0-19	20-99	6	-	-	-	0-19	
45122	1455	0-19	133	548	0-19	-	20-99	-	
451220	1455	0-19	133	548	0-19	-	20-99	-	
…	…	…	…	…	…	…	…	…	
…	…	…	…	…	…	…	…	…	
Total	1166388	15589	98129	190226	7524	5018	22485	12171	etc.

Summary of Evaluation Results

Each method is carried out to the end without having to apply Random Selection method. For cell suppression , there are 9979 (59% of total) cells, or 4,535 (7.5% of total) establishments and 162,368 (14% of total) of employment value that are completely suppressed. With the Systematic Selection method, as shown in Table 4, the entire publication table is successfully protected at the end with only two additional iterations beyond the traditional secondary suppression stage. However the number of cells published in FIs is quite large, not surprisingly since the procedure incorporates the existing secondary suppression methods. For the Systematic Selection method, 10,199 or 60% of all publication cells are selected for FI publication, they account for 6,337 establishments or 10% of all establishments in the table and 180,742 or 15% of total employment in the table. However, In terms of number of publication cells suppressed versus the number of establishments and total values in FIs, the difference between FIPP solution and complete suppression solution is not very large.

Separately for SSSP method and Random Selection method, see summary in Table 5, there are about 64% and 69% of all cells are in FIs respectively. In term of number of iterations required to reach complete protection of the publication table, Systematic Selection takes 2, SSSP takes 3 and Random Selection takes 5. The reason for the difference in the number of iterations could be attributed to the relatively inefficient methods of picking addition PC cells used by the latter two methods. In particular, the Random Selection method does not taken into consideration of the magnitude of the qualified cells. For the total number of additional cells selected beyond the initial stage, SSSP and Random Selection methods selects more cells as FIs than the Systematic method, though the Random Selection method selects significantly more. Table 6 summarizes and compares the total number of cells and values contained in FIs under each of the three different selection methods.

Table 4. FIPP selection-improvement cycle progression – the Systematic method

Publication Table with p=15%, threshold=3

Status	Number of cells	Number of establishments	Employment Level
Safe	8,845 (52%)	59,440 (98%)	1,095,071 (94%)
Primary	8,033 (48%)	1,405 (2.3%)	71,317 (6.1%)
Secondary	0 (0%)	0 (0%)	0 (0%)
Total	*16,878 (100%)*	*60,845 (100%)*	*1,166,388 (100%)*

Secondary Suppression Network Method

Status	Number of cells	Number of establishments	Employment Level
Safe	6,899 (41%)	56,310 (92%)	1,004,020 (86%)
Primary	8,615 (51%)	1,755 (2.9%)	86,358 (7.4%)
Secondary	1,364 (8.1%)	2,780 (4.6%)	76,010 (6.5%)
Total	*16,878 (100%)*	*60,845 (100%)*	*1,166,388 (100%)*

FI publication -- Iteration 0: Publication Cells

Status	Number of cells	Number of establishments	Employment Level
Safe	6,899 (41%)	56,310 (92%)	1,004,020 (86%)
Primary	8,615 (51%)	1,755 (2.9%)	86,358 (7.4%)
Secondary	1,364 (8.1%)	2,780 (4.6%)	76,010 (6.5%)
Exposure	126 (.75%)	725 (1.2%)	7,982 (0.68%)
Total in FI	9,979 (59%)	4,535 (7.5%)	162,368 (14%)
Total	*16,878 (100%)*	*60,845 (100%)*	*1,166,388 (100%)*

FI publication -- Iteration 1 : Publication Cells -- Systematic Method

Status	Number of cells	Number of establishments	Employment Level
Safe	6,789 (40%)	54,310 (90%)	987,488 (85%)
Primary	8,615 (51%)	1,755 (2.9%)	86,358 (7.4%)
Secondary	1,364 (8.1%)	2,780 (4.6%)	76,010 (6.5%)
Exposure	7 (.04%)	45 (.07%)	520 (.04%)
Add'l in FI	210 (1.2%)	1,605 (2.6%)	16,532 (1.4%)
Total in FI	10,189 (60%)	6,185 (10%)	178,900 (15%)
Total	*16,878 (100%)*	*60,845 (100%)*	*1,166,388 (100%)*

FI publication -- Iteration 2: Publication Cells -- Systematic Method

Status	Number of cells	Number of establishments	Employment Level
Safe	6,779 (40%)	54,553 (90%)	985,646 (85%)
Primary	8,615 (51%)	1,755 (2.9%)	86,358 (7.4%)
Secondary	1,364 (8.1%)	2,780 (4.6%)	76,010 (6.5%)
Exposure	0 (0%)	0 (0%)	0 (0%)
Add'l in FI	10 (.05%)	152 (.25%)	1,824 (.16%)
Total in FI	10,199 (60%)	6,337 (10%)	180,724 (15%)
Total	*16,878 (100%)*	*60,845 (100%)*	*1,166,388 (100%)*

Table 5. FIPP selection-improvement convergence pattern comparison between three methods

Iteration	*Total Publication Cells*	*Systematic Selection*			*SSSP Selection*			*Random Selection*		
		Cells with Exposure Risk	*Cells in FIs*	*Add'l PCs Selected*	*Cells with Exposure Risk*	*Cells in FIs*	*Add'l PCs Selected*	*Cells with Exposure Risk*	*Cells in FIs*	*Add'l PCs Selected*
0	16878 (100%)	126 (.75%)	9979 (59%)	-	126 (.75%)	9979 (59%)	-	126 (.75%)	9979 (59%)	-
1	16878 (100%)	7 (.04%)	10189 (60%)	210	15	10495 (62%)	516	75	11004 (65%)	1225
2	16878 (100%)	0	10199 (60%)	10	3	10721 (64%)	226	34	11381 (67%)	377
3	16878 (100%)				0	10772 (64%)	51	14	11525 (68%)	144
4	16878 (100%)							2	11600 (69%)	75
5	16878 (100%)							0	11615 (69%)	15

Table 6. Cells published as FIs by three difference selection methods compared to CSP method

	Systematic	*SSSP*	*Random*	*Cell Suppression*
Number of iterations to reach convergence	2	3	5	NA
Total number of cells in Fis or completely suppressed	10,199 (60%)	10,772 (64%)	11,615 (69%)	9979 (59%)
Total employment level in FIs or completely suppressed	180,724 (15%)	184289 (17%)	188,955 (16%)	162,368 (14%)
Total number of establishments in Fis or completely suppressed	6,337 (10%)	7,362 (12%)	7,971 (13%)	4535 (7.5%)

Conclusion

We developed this FIPP solution with the goal to minimize either the total number of cells selected or the total value contained in the cells selected to be FIs. However because the initial step is built upon secondary cells suppression through CSP solutions and subsequent ad hoc PC selection steps, we probably do not achieve this goal. The good news is that all confidentiality rules imposed on the table are well preserved and the number of cells released as FIs is reasonable at the conclusion of the algorithm. The last audit step on the table clearly demonstrates all primary cells on the table are well protected. With reasonable effort a feasible solution can be found to a seemingly unsolvable optimization problem. The success of this method relies on the assumption that the number of iteration cycles is not large, since current CSP solutions tend to over suppress in the first place. Even with the least inefficient selection method, the Random Selection method, only a maximum of five cycles are needed. The complexity of programming, computer usage time and manual intervention varies depending on selection method used. We found the Random Selection takes the least amount of programming time and manual intervention, SSSP takes longer to run on computer and needs more overhead programming effort, and Systematic method requires more manual interaction during the process than any of the other two methods, therefore is the most cumbersome to use. It is possible with more effort put into the computer programming in the future, we can integrated various parts of the software tasks into a single program. This is necessary if our proposal is to be adopted in regular publication production environment.

One other advantage of our method is that a user can specify cells that he or she does not want to be published in FIs. Once specified, these cells will be treated as if they are constants in the model. The method also allows a user do global coding, i.e. combining categorical variables such that the result will be a table with fewer unsafe cells, though this may need to be done before the selection-audit cycles begin.

We also noticed the following problems with our methods during evaluation of the test data:

1. For the Systematic and SSSP selection methods, the order of the exposure primary cells during each iteration affect the additional PCs selected. In other words, the final set of FI cells could possibly be different if the process is run more than once, since the order of exposure primary cells entered the local protection cycle may be in a different order. Unless the order of cell entry is fixed, which is possible, the process is not repeatable.
2. The Random Selection method produces a different set of selection cells every time it is run, due to the random nature of its selection of PCs in local cycles. This process is not repeatable under this method.
3. Though in theory the methods applies to table with any dimensions and hierarchical structures, as long as the additive relationships in the table is constructible, the computing capability we have so far limit ourselves to only 2-dimentional tables with hierarchical structure in one dimension. Higher dimensional tables require us decomposing the table into lower dimensional tables and process lower dimensional tables separately then integrate separate results at the end. We chose not to experiment that in this study.

Since the test data we used in this study are in reality published as multi-dimensional tables, there are other additive relationships in the table we actually did not study. We only used a two-dimensional marginal table to do the evaluation. Indubitably more cells will be published in FIs and the programming working will be more demanding if the multi-dimensionality is taken into consideration. This is stated in the limitation 3 above. With many other practical issues with publishing cell in FIs not studied here, more works have to be done before we can adopt this method for QCEW data.

Nevertheless, this paper demonstrates with actual data set that our method provides one feasible solution to a seemingly difficult problem. We think our SDC method has good potential for future use in tabular statistical publications.

References

Berkelaar, M., J. Dirks, et al. (2005). The lp_solve software v5.1. Internet download at http://groups.yahoo.com/groups/lp_solve.

Castro, J. (2001). "Using Modeling Languages for the Complementary Suppression Problem Through Network Flow Models." Joint ECE/Eurostat Work Session on Statistical Data Confidentiality.

Castro, J. and N. Nabona (1996). "An Implementation of Linear and Nonlinear Multi-commodity Network Flows." European Journal of Operational Research **92**: 37-53.

Chartrand, G. (1977). Introductory Graph Theory, Dover Publications, Inc.

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

Cox, L. H. (1981). "Linear Sensitivity Measures in Statistical Disclosure Control." Journal of Planning and Inference **5**: 153-164.

Cox, L. H. (1995). "Network Models for Complementary Cell Suppressions." Journal of the American Statistical Association **90**: 1453-1462.

Dula, J. H., J. T. Fagan, and P.B. Massell (2004). "Tabular Statistical Disclosure Control: Optimization Techniques in Suppression and Controlled Tabular Adjustment." Census Research Report Series (Statistics #2004-04).

Fischetti, M. and J. J. Salazar-Gonzales (2000). "Models and Algorithms for Optimizing Cell Suppression Problems in Tabular Data with Linear Constraints." Journal of the American Statistical Association **95**: 916-928.

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

Fischetti, M. and J. J. Salazar (2003). "Partial Cell Suppression: A New Methodology for Statistical Disclosure Control." Statistics and Computing **13**: 13-21.

Giessing, S. (2001). Nonperturbative Disclosure Control Methods for Tabular Data. Confidentiality, Disclosure, and Data Access: Theory and Practical Applications for Statistical Agencies. Doyle, Lane, Theeuwes and Zayatz, North-Holland.

Goodaire, E. G. and M. M. Parmenter (1998). Discrete Mathematics with Graph Theory, Prentice Hall.

Hundepool, A. J., L. C. R. J. Willenborg, et al. (2004). Tau-Argus User's Manual, Version 3.2.

Huo, H. W. (2004). Exercises & Solutions on Algorithms. Beijing, China, China Higher Education Press.

Kao, M. Y. (1996). "Data Security Equals Graph Connectivity." SIAM Journal on Computing **9**: 87-100.

Kelly, J. P. (1990). Confidentiality Protection in Two and Three-Dimensional Tables. College Park, Maryland, University of Maryland, College Park, Maryland. Ph.D. Thesis.

Kelly, J. P., B. L. Golden, et al. (1992). "Cell Suppression: Disclosure Protection for Sensitive Tabular Data." Networks **22**: 28-55.

Langohr, K. and G. Gomez (2005). "Likelihood Maximization Using Web-Based Optimization Tools: A Short Tutorial." The American Statistician **59** (Number 2): 192-202.

Repsilber, R. D. (1994). Preservation of Confidentiality in Aggregated Data. Second International Seminar on Statistical Confidentiality. Luxembourg.

Salazar, J. J. (2001). "Improving Cell Suppression in Statistical Disclosure Control." Joint ECE/Eurostat Work Session on Statistical Data Confidentiality.

Willenborg, L. and T. de Waal (1996). Statistical Disclosure Control in Practice. New York, Springer-Verlag.