Cell Suppression Problem Formulations - Exact Solution and Heuristics

Jim Fagan

August 16, 2001

1 Introduction

Consider a system of n non-negative variables subject to m linear equality "shaft" constraints. This system may be specified by

$$\begin{aligned} Hy &= 0 \\ y &\geq 0\,, \end{aligned}$$

where H is an m by n $\{0, +1, -1\}$ matrix, the columns of which correspond to the cells of a "table", and the rows of which correspond to the shaft constraints of the table. Neither cells which are structural zeros nor shafts which are structural zeros are included in the system. The set vectors y satisfying the above system we will call the set of *feasible* tables.

Let $a = \{a_i\}_{i=1}^{n}$ be the original specified feasible table. Let $P = \{i_1, \cdots, i_p\} \subset \{1, \cdots, n\}$ be the set of p primary suppressions. For each $i_k \in P$, there is a lower bound requirement l_{i_k} and an upper bound requirement u_{i_k}. The aim of suppression is to find, for each k $(k = 1, \cdots, p)$, feasible tables y and z such that

1. $y_{i_k} = a_{i_k} + u_{i_k}$
2. $z_{i_k} = a_{i_k} - l_{i_k}$
3. $y_j = z_j = a_j,\ j \notin P \cup C$, with the set of cells C of complementary suppressions being as "small" as possible.

2 Exact Solution

The above requirements translate exactly to the following mixed integer programming problem (MIP) in variables $(x; y^{(k)}; z^{(k)},\ k = 1, \cdots, p)$. Let $T >> 0$ be a large positive constant.

$$\min \quad c^T x$$

$$\text{s.t.}$$

$$a_i - a_i x_i \le y_i^{(k)} \le a_i + a_i x_i T, \quad i = 1, \cdots, n; \quad k = 1, \cdots, p$$

$$a_i - a_i x_i \le z_i^{(k)} \le a_i + a_i x_i T, \quad i = 1, \cdots, n; \quad k = 1, \cdots, p$$

$$y_{i_k}^{(k)} = a_{i_k} + u_{i_k}, \quad k = 1, \cdots, p$$

$$z_{i_k}^{(k)} = a_{i_k} - l_{i_k}, \quad k = 1, \cdots, p$$

$$H y^{(k)} = 0, \quad k = 1, \cdots, p$$

$$H z^{(k)} = 0, \quad k = 1, \cdots, p$$

$$x_i = 1, \ i \in P$$

$$x \in \{0,1\}^n; \ y^{(k)} \ge 0; \ z^{(k)} \ge 0, \quad k = 1, \cdots, p$$

The choice of cost function c is subjective. For example, if it is desired that the total *number* of cells in C is minimized we take

$$c_i = \begin{cases} 1 & \text{if } i \notin P \\ 0 & \text{if } i \in P \end{cases}$$

On the other hand if it is desired to minimize the total *value* of cells on C we take

$$c_i = \begin{cases} a_i & \text{if } i \notin P \\ 0 & \text{if } i \in P \end{cases}$$

The latter objective function is more desirable for our purposes. While this problem produces exactly the desired solution it is quite unwieldy in practice, except for very small problems with few primary suppressions (even using CPLEX). Consider the example of a standard n^k table with p primary suppression and no structural zeros. (By a *standard* table, we mean a complete table with no heirarchies.) In the CPLEX implementation we have a MIP with $(2p+1)n^k$ varaiable, n^k of which are binary, and $p(4n^k + 2kn^{k-1})$ constraints. This problem grows quickly!

3 Heuristic 1

Insead of processing all primaries at once, we can process them one at a time. It helps to order the primary cells by their required protection levels from largest to smallest. Then for each primary $i_k,\ k = 1, \cdots, p$ we solve the following MIP in variables (x, y, z). Let $T >> 0$ be a large positive constant.

$$\min \quad c^T x$$

$$\text{s.t.}$$

$$a_i - a_i x_i \le y_i \le a_i + a_i x_i T, \quad i = 1, \cdots, n$$

$$a_i - a_i x_i \le z_i \le a_i + a_i x_i T, \quad i = 1, \cdots, n$$

$$y_{i_k} = a_{i_k} + u_{i_k}$$

$$z_{i_k} = a_{i_k} - l_{i_k}$$

$$Hy = 0$$

$$Hz = 0$$

$$x_i = 1,\ i \in P$$

$$x \in \{0, 1\}^n;\ y \ge 0;\ z \ge 0$$

After solving each of the p mixed integer problems, we update the cost function c and the constraints involving x alone. We do this as follows. If C_k is the minimal set on complementary suppressions obtained after solving problem k, we have upon entering problem $k+1$,

$$c_i = \begin{cases} a_i & \text{if } i \notin P \cup\ (\bigcup_{j=1}^{k}\ C_j) \\ 0 & \text{if } i \in P \cup\ (\bigcup_{j=1}^{k}\ C_j) \end{cases}$$

and the constraint

$$x_i = 1,\ i \in P \cup\ (\bigcup_{j=1}^{k}\ C_j).$$

We proceed with each of the primaries, and when we are done we take

$$C = \bigcup_{k=1}^{p}\ C_k.$$

This solution will satisfy the coverage requirement, but not it is not optimal. In the case of the standard n^k table with p primaries and no structural zeros we solve p MIPs each with $3n^k$ variables, n^k of which are binary, and $4n^k + 2kn^{k-1}$ constraints. This problem is more manageable but can still take a long time on a reasonably large table.

4 Heuristic 2

We can get a further heuristic solution which involves solving only linear programming (LP) problems. It again satisfies the coverage requirements but is not optimal. In fact, it can do no better with regard to optimality than the solution obtained with Heuristic 1. The advantage of this procedure is that it is very fast - due to the fact that it involes only solving LPs. As with Heuristic 1, we order the primaries by their required protection levels from largest to smallest. For each primary suppression $i_k,\ k = 1, \cdots, p$ we solve two LPs in variables (y, z) as follows:

$$\min \ c^T(y+z)$$

$$\text{s.t.}$$

$$z \le a$$

$$H(y-z) = 0$$

$$y_{i_k} = u_{i_k}$$

$$z_{i_k} = 0$$

$$y \ge 0,\ z \ge 0$$

and,

$$\min \ c^T(y+z)$$

$$\text{s.t.}$$

$$z \le a$$

$$H(y-z) = 0$$

$$y_{i_k} = 0$$

$$z_{i_k} = l_{i_k}$$

$$y \ge 0,\ z \ge 0.$$

After solving each of the $2p$ linear problems, we update the cost function c. We do this as before. If C_k is the minimal set on complementary suppressions obtained after solving problem k, we have upon entering problem $k+1$,

$$c_i = \begin{cases} a_i & \text{if } i \notin P \cup \ (\bigcup_{j=1}^{k}\ C_j) \\ 0 & \text{if } i \in P \cup \ (\bigcup_{j=1}^{k}\ C_j) \end{cases}$$

We proceed with each of the primaries, and when we are done we take

$$C = \bigcup_{k=1}^{2p} C_k.$$

In the case of the standard n^k table with p primaries and no structural zeros we solve $2p$ LPs each with $2n^k$ variables and kn^{k-1} constraints. This procedure does well in practice.

5 Heuristic 3

We can improve upon Heuristic 2 by processing the upper and lower requirements together with a single LP, rather than separately as was done in Heuristic 2. As with the other heuristics, the coverage requirements are satisfied. With regard to optimality, it can do no worse than Heuristic 2 and no better than Heuristic 1.

As before we order the primaries by their required protection levels from largest to smallest. For each primary suppression $i_k,\ k = 1, \cdots, p$ we solve an LP in variables (y, z, w, v) as follows:

$$\min\ \ c^T(y + z + w + v)$$

$$\text{s.t.}$$

$$z \leq a$$

$$v \leq a$$

$$H(y - z) = 0$$

$$H(w - v) = 0$$

$$y_{i_k} = u_{i_k}$$

$$z_{i_k} = 0$$

$$w_{i_k} = 0$$

$$v_{i_k} = l_{i_k}$$

$$y \geq 0,\ z \geq 0\ w \geq 0,\ v \geq 0.$$

After solving each of the p linear problems, we update the cost function c. We do this as before. If C_k is the minimal set on complementary suppressions obtained after solving problem k, we have upon entering problem $k+1$,

$$c_i = \begin{cases} a_i & \text{if } i \notin P \cup \left(\bigcup_{j=1}^{k} C_j\right) \\ 0 & \text{if } i \in P \cup \left(\bigcup_{j=1}^{k} C_j\right) \end{cases}$$

We proceed with each of the primaries, and when we are done we take

$$C = \bigcup_{k=1}^{p} C_k.$$

In the case of the standard n^k table with p primaries and no structural zeros we solve p LPs each with $4n^k$ variables and $2kn^{k-1}$ constraints. This procedure does well in practice, as well.

6 Heuristic 4

Finally, we can improve upon Heuristic 3 by processing all primaries together in a single LP, rather than separately as in Heuristic 3. As with the other heuristics, the coverage requirements are satisfied. With regard to optimality, it can do no worse than Heuristic 3 and of course no better than the Exact Solution.

We solve an LP in variables $(y^{(k)}; z^{(k)}; w^{(k)}; v^{(k)},\ k = 1, \cdots, p)$ as follows:

$$\min \sum_{k=1}^{p} c^T (y^{(k)} + z^{(k)} + w^{(k)} + v^{(k)})$$

s.t.

$$z^{(k)} \le a, \quad k = 1, \cdots, p$$

$$v^{(k)} \le a, \quad k = 1, \cdots, p$$

$$H(y^{(k)} - z^{(k)}) = 0, \quad k = 1, \cdots, p$$

$$H(w^{(k)} - v^{(k)}) = 0, \quad k = 1, \cdots, p$$

$$y^{(k)}_{i_k} = u_{i_k}, \quad k = 1, \cdots, p$$

$$z^{(k)}_{i_k} = 0, \quad k = 1, \cdots, p$$

$$w^{(k)}_{i_k} = 0, \quad k = 1, \cdots, p$$

$$v_{i_k}^{(k)} = l_{i_k}, \quad k = 1, \cdots, p$$

$$y^{(k)} \geq 0,\ z^{(k)} \geq 0\ w^{(k)} \geq 0,\ v^{(k)} \geq 0, \quad k = 1, \cdots, p.$$

In the case of the standard n^k table with p primaries and no structural zeros we solve an LP with $4pn^k$ variables and $2pkn^{k-1}$ constraints. With the aid of modern software such as CPLEX, this formulation works well - provided there are not too many primary suppressions.

7 Additional Remarks

After applying Heuristic 1, Heuristic 2, or Heuristic 3 and obtaining a set on complementary suppressions, C, we can possibly improve upon our solution as follows. We can do a "cleanup". That is, we can try to unsuppress cells in C.

For each complementary cell j_k, $(k = 1, \cdots, |C|)$ we can run the p MIPs in Heuristic 1, the $2p$ LPs in Heuristic 2, or the p LPs in Heuristic 3 with the following cost function:

$$c_i = \begin{cases} T & \text{if } i = j_k, \quad (T >> 0) \\ 0 & \text{if } i \in P \cup C \text{ and } i \neq j_k \\ a_i & \text{if } i \notin P \cup C \end{cases}$$

If at any stage the optimal value of the objective function in greater than zero, we conclude that cell j_k cannot be unsuppressed and proceed immediately with looking at cell j_{k+1}. If on the other hand we go all through the MIPs of Heuristic 1 or the LPs of Heuristic 2 or 3, and the optimal value of the objective function is always zero, we conclude that cell j_k can be unsuppressed.

In this fashion we get a *minimal* set of complementary supression - minimal only in the sense that no proper subset will satisfy the coverage requirements.

With advanced technology, computers get faster and faster. As time goes on it will become more and more practical to run the LP in Heuristic 4 (and even the MIP in the Exact Solution) on larger and larger problems.