PEDECIBA Informática
Instituto de Computación – Facultad de Ingeniería
Universidad de la República
Montevideo, Uruguay

Reporte Técnico RT 15-13

A Multi-Tree Committee to Assist Port-of-entry Inspection Decisions

Pablo Romero, Jorge Graneri, Omar Viera

Sandro Moscatelli, Libertad Tansini

2015

A Multi-Tree Committee to Assist Port-of-Entry
Inspection Decisions
ISSN 0797-6410
Reporte Técnico RT 15-13
PEDECIBA
Instituto de Computación – Facultad de Ingeniería
Universidad de la República
Montevideo, Uruguay, 2015

A Multi-Tree Committee to Assist Port-of-Entry Inspection Decisions

Pablo Romero[(a,b)]; Jorge Graneri[(a)]; Omar Viera[(b)]; Sandro Moscatelli[(b)]; Libertad Tansini[(b)]

[(a)] *Laboratorio de Probabilidad y Estadística, Facultad de Ingeniería, Universidad de la República, Montevideo, Uruguay. Julio Herrera y Reissig 565, PC 11300. Tel-Fax (+598) 2711 0621.*
Email: promero@fing.edu.uy, jgraneri@yahoo.fr
[(b)] *Instituto de Computación, Facultad de Ingeniería,*
Universidad de la República, Montevideo, Uruguay.
Julio Herrera y Reissig 565, PC 11300. Tel-Fax (+598) 27114244.
Email: {viera,moscatel,libertad}@fing.edu.uy.

Abstract

A natural way to avoid the injection of potentially dangerous or illicit products in a certain country is by means of protection, following a strict port-of-entry inspection policy. A naive exhaustive manual inspection is the most secure policy. However, the number of within containers allows only to check a limited number of containers by day. As a consequence, a smart port-of-entry selection policy must trade cost of inspection with security, in order to fit into the dynamic operation of a port.

We explore the design of port-of-entry container inspection policies with imperfect information (unavailable or untrusted data). Starting from an a-priori classification provided by port-of-entry customs operator, a combinatorial optimization problem is introduced. The goal is to match an a-priori container classification with a logically coherent one, subject to a given level of container inspection. Inspired in the related literature, a novel Multi-Tree committee is introduced in order to find a solution to the previous combinatorial problem. It combines the strength of binary decision trees and minimization of logical functions. The algorithm is easy-to-handle and useful for an on-line production. We highlight the effectiveness of our proposal, regarding real traces available from the port of Montevideo. The results show the capability to detect the most risky containers and its conservative nature, respecting any desired level of inspection.

Preprint submitted to European Journal of Operational Research *July 19, 2015*

Keywords: Risk analysis, Port-of-Entry, Inspection Policy, Binary Decision Tree, Multi-tree Committee.

1. Introduction

The decisions ruled by a port-of-entry customs operator must overcome several challenges, as the presence of illicit products or even foreign attacks. The source of inspiration is their experience to detect anomalies and sometimes, it is assisted with electrical detection by sensors and operational research-based algorithms for manual inspection.

The real problem is so complex that theoretical tools and automatic algorithms usually assist, but do not replace, the decision of an operator. The main difficulties arise when the available information is untrusted, incomplete or imperfect. Furthermore, there is no international agreement in the encoding of information. As a consequence, an automatic approach is highly desirable, but it is hard to design. Moreover, it is difficult to compare the sensibility of different models against the presence of suspicious containers. So far, the proposed approaches from the operational research community ranges from game-theoretic models [17], design of logical functions [20], binary decision trees [13, 16], multi-objective optimization [19, 21], dynamic programming techniques [21], and Bayesian approach [4], among many other ad-hoc models for screening under budget constraints [7] and prophylactic models for terrorist attacks [14].

In this paper, we propose a novel approach with the objective of "correcting" inconsistencies found within the a-priori classification provided by port-of-entry customs operators according to their work experience. The decisions are supported by a set of rules or risk-indicators, that suggest which containers are suspicious and which are not. The goal is to enforce coherence in the classification, i.e. a container with more risk-indicators should be classified as more suspicious, and therefore should be pointed out to be manually checked with higher probability. As far as we know, the detection and correction of inconsistencies in a real-life port-of-entry inspection policy has not been addressed in previous works in the related literature. The main contributions of this paper are summarized in the following items:

- To the best of our knowledge, this is the first paper in the area that enriches an operational research approach based on previous experience offered by customs operators in a real port.

- Previous works assume perfect knowledge of detection/false alarm probabilities, or percentage of illicit containers [4, 13, 20, 16, 1, 2]. However, a pointwise estimation of those probabilities are usually performed with small sample size in relation with the whole container population. We believe this assumptions lead to unaccurate models. In this sense, our model seems to be more *realistic*.

- We provide an explicit formulation of a combinatorial optimization problem, called Coherence Problem. The goal is to enrich an a-priori classification given by real customs operators, adding coherence and respecting a desired level of inspection (i.e. the fraction of containers to be inspected).

- A novel Multi-Tree Committee is introduced in order to provide a feasible solution. It combines the strength of Binary Decision Trees (BDTs) and minimization of logical functions.

- We obtained real traces offered by our National Port Operator, called Administración Nacional de Puertos, located at Montevideo, Uruguay. We will appreciate the capability of our Committee-based approach to detect the most risky containers, and its conservative nature, respecting the level of inspection imposed by customs operators.

This document is organized as follows. Section 2 contains related work, mainly focused on binary decision trees and logical design of inspection policies. In Section 3, an ideal approach is developed. Here, we try to follow the classification provided by customs operators as close as possible. A Mirror solution presents the minimum gap. However, it is not coherent, in the sense that more risky containers are not opened. Inspired in the deficiencies of the previous approach, a Realistic Approach is presented in Section 4. It captures a notion of coherence in the decision which is usually not met in real-life, giving possibilities of enhancement and automation. Furthermore, a desired fraction of containers is selected to be inspected, hence it achieves a realistic level of inspection.

In Section 5, an algorithm called *TreeCommittee* is introduced for the general Coherence Problem. It tries to unify the strength of both worlds: binary decision trees and logical functions. As we will see, the role of the committee is to cope with imperfect information and to adjust to a desired level of inspection. Section 6 illustrates the results on the lights of real traces from our national port of Montevideo. There, we study the gap between $TreeCommittee$ and $Mirror$. Finally, in Section 7 we point-out concluding remarks and discuss trends for future work.

2. Related Work

The inspection process can be regarded as the collection and analysis of information available from multiple sensors and other sources, in order to decide whether a container should be allowed to enter the port or not [20].

The means to detect anomalies are supported in digital signal processing of physical information of containers and consistency of the documentation related with the container. J. Candy provides an X-ray physics-based threat detection solution, using a probabilistic model [4]. The Bayesian approach enables the development of a physics-based detection algorithm capable of detect threat under restricted assumptions, including known detection and false alarm probabilities.

David Madigan, Sushil Mittal and Fred Roberts develop an abstraction of the port-of-entry inspection problem [13], assuming the existence of n logical sensors which return binary variables $\{d_i\}_{i=1,\ldots,n}$, such that $d_i = 1$ means that the feature number i studied by that sensor is suspicious; and $d_i = 0$ otherwise. As a consequence, each container has a corresponding binary word $d = d_1 \ldots d_n$, and the authors study classification mechanism of binary words by means of binary decision trees (BDT). A BDT is a widely used classification technique, where there is a special root-node and all nodes have output-degree 2. The container information i is inserted in the root-node, which is identified with a question (precisely, one sensor labeled with the numbers $\{1, \ldots, n\}$). If the answer is affirmative (negative), the information i travels to the right (left) branch of that node. Finally, there is a correspondence between the terminal nodes and the set $\{0, 1\}$, in order to assign a binary classification to every container. A challenging problem is to design the questions to be inserted in each node. As a consequence, the design of a BDT is reduced to a sequential decision problem, where the aim is to decide the order of the sensors to minimize a cost function.

A related perspective of the problem is to design a binary function $f : \{0, 1\}^n \rightarrow \{0, 1\}$, where the containers are manually classified whenever their corresponding binary word d complies $f(d) = 1$. There are several realizations of a certain binary function $f(d)$ with equivalent BDTs. Laurent Hyafil and Ronald Rivest proved in 1976 that the minimization of the expected number of questions in the set of equivalent BDTs is an NP-Complete problem, where the probabilities $P(d_i = 1)$ are known [11]. In the year 2003, Phillip Stroud [16] provides an exhaustive enumeration of BDTs such that no sensor appears twice in any branch. Specifically, if N_n is the number of such BDTs, then $N_n = 2 + n(N_{n-1})^2$, where $N_0 = 2$ (with no sensor, the only BDTs are either output 0 or 1). The reader can easily check that there are only 74 BDTs with $n = 2$ sensors, but $N_5 \approx 5 \times 10^{18}$. The cardinality N_n of BDTs in terms of the number of sensors n has an exponential growth, and an exhaustive evaluation of BDTs is not practical for a large number of sensors. Stroud defines the cost of a tree combining both the misclassification probability and inspection cost (i.e. the use of sensors). The first contribution depends on the failure probability of the sensors, as well as the statistical distribution of the containers, that must be previously estimated from historical information. It sounds mandatory to reduce the search space. Therefore, the author finds an optimal BDT among a reduce subset. They ranked all trees formed from 3 or 4 sensors according to increasing tree costs, and defined an enumeration algorithm. Their results can not be directly extended to higher number of sensors. However, Anand et. al point out that the optimal inspection policy provided by Stroud algorithm is remarkably insensitive to parametric changes [1], reinforcing their approach.

Similar in spirit and with a BDT-based classification, David Madigan first tunes the thresholds for each sensor in order to minimize the cost function, by means of numerical methods for non-linear equations, combining gradient-descent and Newton-Raphson. A key element of his development is to study a distinguished sub-class of BDT, which are monotonous and complete (a BDT is monotonous when it realizes a monotonous binary function, and complete whenever the BDT includes all sensors). In this sub-world of BDT, an algorithm for exhaustive enumeration is feasible, and the author finds the optimal BDT when $n = 4$. The number of BDTs in this subset is now $263,515,920$ when $n = 5$. The number of monotonous logical functions with n bits is called Dedekind's number since his seminal work in 1897 [6], and the exact number is an open problem for large number of n (recent advances in this area confirm for instance the case $n = 8$ [18]).

The search in the space of complete-monotonous BDTs takes into consideration a neighborhood structure (i.e. a polytime and completely transitive neighborhood system) that swaps, deletes, merges and replaces nodes for cost minimization. Nevertheless, the multi-modal nature of the objective function leads the local search to get stuck in local minima. As a consequence

Madigan introduces a heuristic approach with randomness, inspired in simulated annealing. The heuristic is tested with the set of complete monotonous BDT, reaching 42 times the global optimum, which was more economic than the BDT found by Anand, who search even in the whole universe of BDT. Since the complete study for $n = 5$ also escapes to million of tree configurations, it was suggested as a trend for future work, and suggests the use of genetic algorithms. Indeed, the design of genetic algorithms to face a multimodal objective has been suggested in other works [12, 9], and implemented in subsequent works [15, 5].

More recently, Christina M. Young tackles the port-of-entry inspection problem in order to avoid undesired cargo in the United States, in her PhD thesis [20]. The thesis includes a background of the problem and discussion of several formulations for the port-of-entry inspection problem by mathematical models, with a neat treatment of the assumptions and horizon of applicability. Based on Elsayed et. al [8], the goal of the models is to reduce the overall expected cost in the space of sequential inspection policies. Following works from Stroud and Saeger [16], S. Anand [1] and Boros [2], it is assumed that the sensors return normally distributed numerical values. Christina M. Young finds optimal thresholds by numerical estimation, that lead to a binary output. A sequential configuration of the sensors can be identified with parallel-series logical circuits or equivalently, conjunctive normal forms. Let π be the proportion of containers with risky elements, that should be manually inspected under optimal conditions, and let us denote with X the random variable that assumes the value 1 whenever the container has illegal cargo and 0 otherwise: $\pi = P(X = 1)$. Consider $p_i = P(d_i = 1)$ and their complements $q_i = 1 - p_i$. Additionally, the utilization of sensor i has a cost c_i.

The most strict policy F_s will define a container as risky whenever there is at least one sensor set to 1. In normal form it is $F_s = \vee_{i=1}^{n}\{d_i = 1\}$. The following question has a simple answer, but illustrates a valuable concept for the problem under study. Which is the optimum sequence for the container inspection under strict policy? The answer is to order the sensors with low cost and higher sensibility to positive first: $c_1/q_1 < c_2/q_2 < \ldots < c_n/q_n$, where we re-labeled the sensors to respect the previous inequalities.

The opposite policy is the most permissive, and defines a container as risky only when all sensors are positive: $F_p = \wedge_{i=1}^{n}\{d_i = 1\}$. Which is the optimal sequence in this case? In this case, the sensors with highest sensibility to negative should be revised before, weighting their costs. More precisely, the order must respect $c_1/p_1 < c_2/p_2 < \ldots < c_n/p_n$.

The previous results are intuitive, and were proved for the first time by Zhang et. al and further generalized to all logical functions [20, 8]. It is worth noting that the correct use of this optimization tool requires knowledge about the probabilities $p_i = P(d_i = 1)$ for each sensor, as well as the proportion π of risky containers and the cost of utilization c_i of all sensors. Christina M. Young also weights the cost of inspection in the cost function by a convex combination between misclassification and inspection, or Pareto curves that trade both aspects as desired.

An alternative approach is to exploit historical classification provided by port operators and conduct a mimetic solution. In [10], we provide the closest deterministic classification to that offered by customs operators. The measure of similarity is there captured by the number of matchings in the classification of a part of the training set, and the result has a statistical meaning.

This work is widely inspired in prior works from David Madigan and Christina M. Young [20]. However, the context is rather hostile, since the information available is not statistically significant to define correct estimations for the misclassification probabilities, nor the proportion of risky containers π. As pointed out before, the novelty of this approach is the introduction of consistency to a previously given incoherent classification. Additionally, we will combine the strength of BDTs and minimization of normal forms evaluation in order to define an inspection policy suitable for an arbitrary number of sensors and level of inspection. In Section 3, we conduct an ideal approach, and a summary of the main ingredients provided in [10]. Then, a realistic analysis is carried out in Section 4. A corresponding realistic classification is presented in Section 5, and a later comparison between both approaches is highlighted in Section 6.

3. Ideal Approach

We will follow the terminology from Section 2, where we are given a training multiset $W = \{w_1, \ldots, w_N\}$ of (possibly repeated) binary words, for N containers. The set $\{1, 2, \ldots, r\}$ will be denoted by $[r]$ for short, and represent the levels of risk. Each container is labelled (by customs operators) a risk level $l_1, \ldots, l_N$, where $l_i \in [r]$. In the real problem, we are also concerned with a level of inspection α, but we will not study this constraint until the following section. In a first ideal approach, the goal is to define a deterministic function $f : \{0,1\}^n \to [r]$ that automatically classifies containers, and is *similar* to the labels provided by customs operator. This approach reflects the expertise from customs operators, where the fuzzy classification is realized in an automatic manner.

Observe that we do not assume misclassification probabilities nor sensor calibration. Therefore, to the best of our knowledge, there is only one prior work in the area that is suitable to the conditions of our real problem, written by the same authors [10]. This article is our point of departure, and will be summarized in this section as an ideal approach.

Two notions of similarity are considered. The symmetric difference between f and the multiset W of binary words is:

$$\Delta_W(f) = \sum_{i=1}^{N} 1_{\{f(w_i) \neq l_i\}}$$

An alternative measure of similarity is the mean square error:

$$d_W(f) = \frac{1}{N}\sum_{i=1}^{N}(\{f(w_i) - l_i\})^2$$

The following combinatorial optimization problem formalizes the Ideal Approach:

$$\min_{f:\{0,1\}^n \to [r]} d(f, W),$$

being either $d(f, W) = \Delta_W(f)$ the symmetric difference or $d(f, W) = d_W(f)$ the mean square error.

It is well known from elementary calculus that the mean square error is minimized when $f(w)$ is an averaging and rounding among all the labels $w_i \in W$ such that $w_i = w$. Let us denote f_1 to that average-and-rounding classification rule (for convention, we choose $f_1(w) = 1$ whenever $w \notin W$). Now, let us consider $d(f, W) = \Delta_W(f)$, what is called the Decision Problem. The following definition has been introduced by the same authors in [10]:

Definition 1. *Given the multiset $W = \{w_1, \ldots, w_N\}$ and respective labels $l_1, \ldots, l_N \in [r]$, the* Mirror Solution *is the function $f_2 : \{0,1\}^m \to [r]$ where $f_2(w)$ is the empirical mode (i.e., the most frequent label for w in the multiset W). As a convention, we choose $f_2(w) = 1$ whenever $w \notin W$.*

The following result is intuitive.

Proposition 1. *The Mirror Solution is a global optimum for the Decision Problem.*

Proof. Let us consider an arbitrary solution $g : \{0,1\}^m \to \{1,\dots,k\}$ and the Mirror Solution f_2. We will establish the inequality $\Delta_W(f_2) \leq \Delta_W(g)$.

Let us consider the frequency of the binary word w in the multiset W with label i, this is $t_i(w) = \{j \in [1,N] : w_j = w,\, l(w_j) = i\}$.
The number of incorrect assignments to word w using f_2 is precisely
$\varepsilon(f_2, w) = (\sum_{1\leq i\leq k} |t_i(w)|) - \max\{|t_i(w)| : 1 \leq i \leq k\}$. If $g(w) \neq f_2(w)$ then the number of incorrect assignments to word w using g is
$\varepsilon(g, w) = (\sum_{i=1}^{k} |t_i(w)|) - \sum_{i=1}^{k} 1_{\{g(w)=i\}}|t_i(w)|$.
Since only one of those indicators is true, we immediately conclude that $\varepsilon(f_2, w) \leq \varepsilon(g, w)$, and

$$\begin{aligned}
\Delta_W(f_2) &= \sum_{i=1}^{N} 1_{\{f_2(w_i)\neq l_i\}} \\
&= \sum_{w\in\{0,1\}^m} 1_{\{w\in W\}}(\sum_{i=1}^{k} |t_i(w)|) - \max\{|t_i(w)|, i = 1,\dots,k\} \\
&= \sum_{w\in\{0,1\}^m} 1_{\{w\in W\}}\varepsilon(f_2, w) \leq \sum_{w\in\{0,1\}^m} 1_{\{w\in W\}}\varepsilon(g, w) = \Delta(g, L).
\end{aligned}$$

□

For the sake of simplicity, we will consider the deterministic rule f_1 (average-and-rounding) as the classification provided by customs operators. It is the optimal rule in the sense that it minimizes the mean square error.

4. Realistic Approach

We we follow the terminology from Section 3. We are given a triad (W, f, α), where $W = \{w_1, \dots, w_N\}$ represents a multiset of (possibly repeated) binary words, $f = f_1$ is the averaging-rounding rule, that assigns a risk-value in the set $[r]$ to each container, and a level of inspection $\alpha \in [0,1]$ that determines the fraction of containers to be inspected.

The project signed by our university and the port pursues the goal of designing an algorithm that should adapt to a variable number of sensors n

and risk-levels r, meeting at the same time a desired level of inspection.

Observe that the set of binary words accepts a partial order: if $w = d_1 \ldots d_n$ and $w' = d'_1 \ldots d'_n$, then $w \leq w'$ if and only if $d_i \leq d'_i$ for all $i = 1, \ldots, n$. A function $g : \{0,1\}^n \to [r]$ is monotonous, or *coherent*, if $g(w) \leq g(w')$ whenever $w \leq w'$. In practice, the classification f offered by the customs operator is *not* coherent. Indeed, there are certain containers that were assigned a low classification, while they should have been classified as risky, in terms of sensors.

Therefore, we are motivated to introduce the Coherence Problem. Let us denote $\Delta(f,g)$ the symmetric difference between two given functions f and g. More precisely, if $f : \{0,1\}^n \to [r]$ and $g : \{0,1\}^n \to [r]$ then

$$\Delta(f,g) = \sum_{w \in \{0,1\}^n} 1_{\{f(w) \neq g(w)\}},$$

where $1_{\{x\}}$ equals 1 if x is true, and 0 otherwise. Alternatively, the symmetric difference with respect to the multiset W of binary words is:

$$\Delta_W(f,g) = \sum_{i=1}^{N} 1_{\{f(w_i) \neq g(w_i)\}}$$

Our goal is to find the coherent classification g that is closest to the given one f, with respect to the multiset W, respecting a given level of inspection $\alpha \in (0,1)$, that determines the fraction of containers to be inspected. The Coherence Problem is formulated as follows:

$$\min_{g,I} \Delta_W(f,g)$$

$$s.t.$$

$$f(w) \leq f(w'), \forall\, w \leq w' \tag{1}$$

$$\sum_{w \in W} 1_{\{g(w) > I\}} \leq \alpha |W| \tag{2}$$

$$I \in [r] \tag{3}$$

In words, coherence is introduced to the classification f provided by the operator (Constraint 1), meeting at the same time a certain level of inspection α (Constraint 2). Decision variable I plays the role of a threshold, where the containers to be inspected w must respect the inequality $g(w) > I$. Constraint 3 states that the threshold is an arbitrary integer in the set $[r]$.

The reader can appreciate that the Coherence Problem does not involve misclassification probabilities, in contrast to most previous works in the area [4, 13, 20, 16, 1, 2].

5. Multi-Tree Committee

In this section we will propose an algorithm to address the Coherence Problem in its most general formulation. Even though it is more sophisticated than averaging or statistical mode, as a result we will have a coherent solution that meets the levels of inspections that must be met in real-life applications.

As stated before, any arbitrary container is identified with a binary word $w = d_1 \dots d_n$, $d_i \in \{0, 1\}$. A classification is a function $g : \{0, 1\}^n \to [r]$. The main reason to return a function g with range in the same set $[r]$ is that the symmetric difference $\Delta_W(f, g)$ can be dynamically compared by new functions f and multisets W offered by customs operators. Furthermore, containers classified on-line can be added to the training data in the future (adding the concept of closed-loop or feedback in our control system). $TreeCommittee$ receives the multiset W, classification $f : \{0, 1\}^n \to [r]$, a level of inspection α and returns a coherent classification g and threshold I.

Algorithm 1 *TreeCommittee*

Inputs:
Multiset: W
Classification: $f : \{0,1\}^n \rightarrow [r]$
Level of Inspection: $\alpha \in [0,1]$
Output:
Coherent Function: $g : \{0,1\}^n \rightarrow [r]$
Threshold: I

```
 1: for $i = 1$ to $r - 1$ do
 2:     $g_i = Convert(W, f, i)$
 3:     $T_i = OptimizeBDT(W, f, g_i)$
 4: end for
 5: $g = 1 + \sum_{i=1}^{n} T_i$
 6: $I_{aux} \leftarrow 1$
 7: $q \leftarrow Count(g(W) > I_{aux})$
 8: $\alpha_{aux} \leftarrow q/|W|$
 9: while $\alpha_{aux} \leq \alpha$ do
10:     $I_{aux} \leftarrow I_{aux} + 1$
11:     $q \leftarrow Count(g(W) > I_{aux})$
12:     $\alpha_{aux} \leftarrow q/|W|$
13: end while
14: $I \leftarrow I_{aux} - 1$
15: return $(g, I)$
```

A key element is to translate the problem into $r-1$ binary sub-problems. This is where it is possible to exploit the knowledge provided from prior works in the field, covering structures of binary decision trees, logics in normal form, heuristics and minimization. Precisely, $Convert(W, f, i)$ receives a set of binary words in its first argument, a function $f : \{0,1\}^n \rightarrow [r]$, a positive integer $i \in \{1, \dots, r-1\}$ and returns the binary function $g_i : W \rightarrow \{0,1\}$ such that $g_i(w) = 1_{\{f(w)>i\}}$ (Line 2). The heart of *TreeCommittee* is Function *OptimizeBDT*, that is applied sequentially to each function g_i during *for* loop (Line 3). It returns a binary decision tree that is equivalent to a coherent function (related with the input function). The output g is precisely a Multi-Tree Committee. It combines the opinion of the tree-committee $\{T_i\}_{i=1,\dots,r-1}$ in a simple fashion: the risk is the number of risky opinions from the trees, or *sum risk*, plus one (Line 5). During the block of Lines 6-14 the threshold

I is found. An auxiliary variable I_{aux} is iteratively increased, until the level of inspection is not met. Then, the correction $I = I_{aux} - 1$ takes place in Line 14. Both decision variables (g, I) are returned correspondingly in Line 15.

Remarks.

1 *The range of g is precisely $[r]$ as desired, since $T_i(w) \in \{0,1\}$, for all $i \in \{1, \ldots, r-1\}$.*

2 *The functions g_i are defined in the multiset W, but in the whole set $\{0,1\}^n$.*

3 *Moreover, functions g_i may be non-coherent.*

4 *A sum of coherent functions is coherent. Therefore, the output committee function g inherits the monotonicity of its terms T_i.*

As a consequence, *OptimizeBDT* should introduce coherence to the input function of its third argument (i.e. g_i), extend its domain and translate it to an equivalent binary decision tree. That tree will be further optimized by means of a local search and rewritten to minimize the number of questions, inspired in parallel-series minimization of logical functions [20].

Algorithm 2 $T = OptimizeBDT(W, f, g)$

```
1: $g^1 = Coherence(g, W)$
2: $g^2 = Complete(g^1)$
3: $T^1 = Translate(g^2)$
4: $T^2 = LocalSearch(W, f, T^1)$
5: $T^3 = SpeedUp(W, T^2)$
6: return $T = T^3$
```

Coherence receives a (possibly incomplete) binary function g and returns a coherent function g^1 with the same domain. Consider the partially ordered set $R = (\{0,1\}^n, \leq)$. *Coherence* uses the relation in the subset W inherited by R, and explores each maximal chain. If during that process there are two words $w \leq w'$ with $g(w) = 1 > g(w') = 0$, then $g(w)$ is set to 0. The process is finite, and the result is a coherent function g^1. Analogously, *Complete* extends g^1 preserving coherence (iteratively in each chain by means of Zorn's Lemma), and returns g^2. If a certain word w not in W has some lower bound set to 1, then $g^2(w) = 1$. If it has an upper bound set to 0, then $g^2(w) = 0$. Otherwise, if w is not comparable with any element in W, we set $g^2(w) = 0$.

Once the complete coherent function g^2 is obtained, a translation to a binary decision tree takes place (Line 3). In order to carry-out the translation, a canonical reduced representation of logical functions is used, by means of an ordered binary decision diagram (OBDD), following recommendations from [3]. Function $Translate$ works in three stages. These stages are illustrated one-by-one with an example (the reader can find properties of OBDDs and a thorough overview in [3]). Consider the input function g from Table 1.

Table 1: Sample binary function g to be translated.

d_1	d_2	d_3	g
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

Figure 1 shows an equivalent complete binary-tree representation of g, where the output values 0 or 1 are written in blocks. Observe that solid external links are used when the sensor returns an affirmative answer, while negative answers are represented by external dashed links. In the second stage, internal nodes are identified whenever they have the same answer (i.e. the same external nodes), as illustrated in Figure 2. Finally, in the third stage redundant questions are deleted (see Figure 3 for the application in the example). These stages are iteratively applied, until no modification is feasible (in the example the process is complete).

A naive local search is included in Line 4, where two questions (nodes) are swapped whenever the solution is both feasible and has lower symmetric difference with the given function.

Finally, a *SpeedUp* process takes effect, inspired in prior work of Christina Young [20]. *SpeedUp* does not change the logical value of the input tree T^2, but just rewrites the sequential order of the questions. Precisely, we write the equivalent logical value for T^2 with a normal form $T^2 = \vee_{i=1}^{m} \left(\wedge_{j=1}^{r} \varphi_{ij} \right)$, where each φ_{ij} assumes a possible value d_k or its negation $\overline{d_k}$ for some $k = 1, \ldots, n$. Both operations of disjunction ($\wedge$) and conjunction ($\vee$) are commutative.

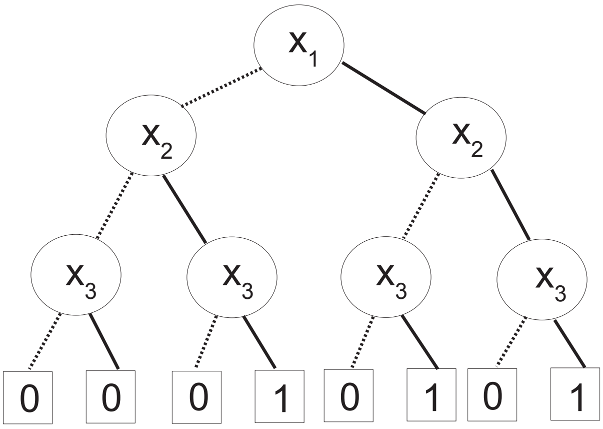

Figure 1: *Translate* - Stage 1: each binary word $w = d_1d_2d_3$ is assigned the value $g(w)$.

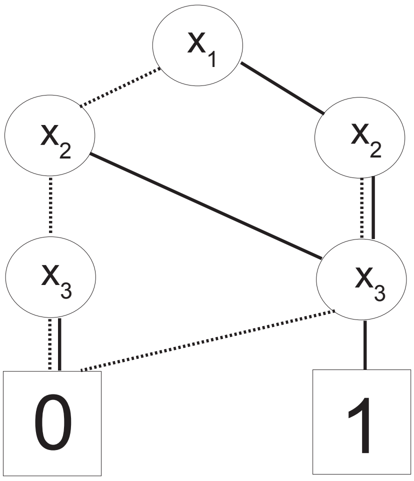

Figure 2: *Translate* - Stage 2: Internal nodes with identical answers are identified.

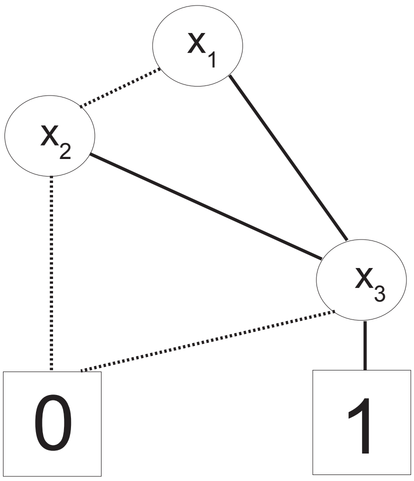

Figure 3: *Translate* - Stage 3: Deletion of redundant questions.

Therefore, we can commute the order and the logical result is unaffected, but the computational effort (for instance, the number of questions) can be reduced. For each sensor, the probabilities $p_i = P(d_i = 1)$ can be estimated with the statistical average, and $q_i = 1 - p_i$. Since in this problem the costs c_i of the sensors are similar, we fix $c_i = 1$. The least frequent clauses of T^2 in the set W (concretely, the less frequent events of the set $\{\wedge_{j=1}^{r} \varphi_{ij}\}_{j=1,\ldots,r}$) appear first, whereas the most frequent literals φ_{ij} of each clause appear first. To summarize, Function *SpeedUp* rewrite the input tree exploiting the fact that the operations $\vee$ and $\wedge$ are commutative, in order to minimize the computational effort in the evaluation.

6. Empirical Results

The customs operators from Administración Nacional de Puertos (ANP) provided us a classified training data of $|W| = 1433$ labelled containers in order to design a port-of-entry inspection algorithm with a level of inspection $\alpha = 0,02$. The labels accept $r = 3$ possibilities according to their expertise, where 1 means low risk, 2 medium risk and 3 represents the highest risk.

We summarize from our analysis from Sections 3 and 5 the following three solutions, and the corresponding terminology:

- Average-and-rounding rule, f_1, is the best inspection policy for the Ideal Approach, when the mean square error is being minimized.
- Mirror solution, f_2, is the best inspection policy for the Decision Problem, when the symmetric difference is being minimized.
- $TreeCommittee$, denoted by g, is a feasible solution for the Coherence Problem, where the solution meets both coherence and a desired level of inspection.

The operators suggested us to use $n = 6$ risk-indicators, that we will label as sensors, using the classical terminology, with output $w = d_1 d_2 \ldots d_6$, being $d_i \in \{0, 1\}$. These sensors take the training data and find the output d_i by means of a statistical analysis, reflecting anomalies with high risk. For the sake of data privacy, the adjustment of the sensors and their thresholds is excluded (in fact, they return numerical values instead of binary ones). However, we followed a traditional optimization technique by means of a discretization of the n-dimensional cube, in order to find thresholds and translate real values into binary ones [10].

As a pre-processing stage, we studied the correlation between each indicator d_i and the given classification. We excluded one indicator from our study, since its output was consistently null for all containers in the training data (as a consequence, a positive answer from this sensor would mean an immediate manual inspection). Additionally, by rank-correlation and independent tests we could identify another indicator that has weak correlation with the given classification. From previous observations, we work with four risk-indicators.

Incidentally, Madigan suggests optimal solutions when $n = 4$ [13]. However, Madigan's methodology is not applicable for this scenario, since we do not have historical information of false positives and negatives in order to estimate the misclassification probability.

Table 2 shows the discrepancy between $Mirror$ solution and the labels provided by customs operators $L = \{l_1, \ldots, l_{1433}\}$. Observe that the main diagonal represents the number of matchings between $Mirror$ and the labels, while the sum of the remaining entries is the symmetric difference.

Table 2: Matchings between $Mirror$ and labels

f_1 / W	1	2	3
1	390	96	31
2	102	581	96
3	18	96	23

Observe that $Mirror$ classifies $390 + 581 + 23 = 994$ containers correctly out of 1433, so $\Delta_W(f_1) = 439$. From Proposition 1, we know that $Mirror$ presents the lowest symmetric difference. However, as an ideal approach, it does not meet coherence nor level of inspection of 2%.

We know that our $TreeCommitee$ solution will present lower matchings than $Mirror$, since the Coherence Problem is multi-constrained (see Table 3 for details). However, in order to have a major understanding of its performance it is worth to have a closer look of its classification in terms of the labels of specific words, coherence and level of inspection.

Table 3: Matchings between $TreeCommittee$ and labels

g/W	1	2	3
1	502	762	142
2	8	11	1
3	0	0	7

We implemented $TreeCommittee$ in Matlab and injected the given classification W with labels, with a running time close to one minute in a home-PC. We obtained the binary trees T_1 and T_2 illustrated in Figures 4 and 5 respectively, together with the output classifier g from Table 4.

The first tree penalizes more cases than the second-one. This is reasonable, since the corresponding input functions g_1 and g_2 of *OptimizeBDT* respect the inequality $g_1 \geq g_2$. This is a strong indication of consistency from the tree-building block, since the level of inspection of T_2 is lower than one of T_1. Constraint 2 states that the number of words $w \in W$ such that $g(w) > 1$ must not exceed $\lfloor \alpha |W| \rfloor = \lfloor 0,02 \times 1433 \rfloor = 28$ elements. The number of containers $w \in W$ such that $g(w) > 1$ are precisely the ones with a positive answer of some binary tree (equivalently, the rows from Table 4 such that $g(w) > 1$), and represents $27/1433 \times 100 = 1,88\%$, respecting the

Table 4: Final classifier g: the last row represents the remaining words $z_1z_2z_3z_4 \in \{0,1\}^4$.

d_1	d_2	d_3	d_4	g
1	1	1	1	3
0	1	1	1	3
1	0	1	1	2
1	0	1	0	2
1	1	0	1	2
1	1	1	0	2
z_1	z_1	z_3	z_4	1

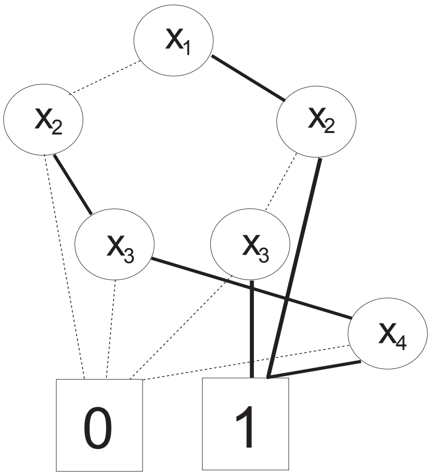

Figure 4: First committee-member T_1 of $TreeCommittee$.

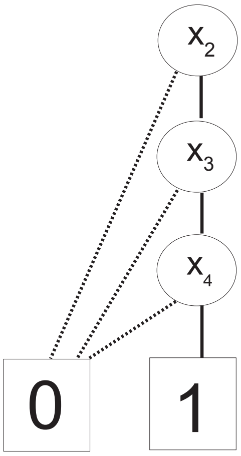

Figure 5: Second committee-member T_2 of $TreeCommittee$.

level of tractability accepted by the dynamism of our national port (fixed in 2%).

There were only seven real containers in the training data that have been classified with maximum risk 3, since those containers were common to both trees, representing a level of inspection of $7/1433 \times 100 = 0,488\%$. According to the coherence-guided process, the most risky word $w = 1111$ is assigned the maximum risk ($g(1111) = 3$), while $w = 0000$ has minimum risk ($g(0000) = 1$). There was exactly one container with maximum risk in the training set, and $f(1111) = 3$ accordingly. There were exactly six containers that present the word $w = 0111$, and respect $g(0111) = f(0111) = 3$.

The words with intermediate classification represent a mass of 20 containers. Both words $w = 1010$ and $w = 1011$ occur exactly in two containers each, and they have been classified with intermediate risk by both inspection policies: $g(1010) = f(1010) = 2$ and $g(1011) = f(1011) = 2$. We appreciate a reasonable matching, regarding our original assumption that the target objective is a logical function. The case with highest gap between the labels and MultiTree Committee occurs in the binary word $w = 1101$. This word appears 16 times in the set W and $g(1011) = 2$, but it was assigned 8 times the maximum risk, 7 times intermediate risk and 1 time the lowest risk, by customs operators. This gap suggests the need to have a higher number of rules. Additionally, the operators wish to include randomness to the current algorithm, in order to try an optimistic manual inspection of "non-risky" containers from time to time.

7. Conclusions

In a real-world port-of-entry inspection problem, misclassification probabilities are hard to estimate, and customs operators use their experience but sometimes lack of a theoretical foundation. However, to the best of our knowledge, prior works assume perfect knowledge of these probabilities, which is not our case. Additionally, our context is hostile, since the training data is inaccurate.

In this article, a Coherence Problem has been introduced, and a Multitree Committee algorithm has been developed in order to define a port-of-entry inspection policy. Both the model and its resolution are deterministic and easy-to-handle, which is a relevant aspect in practice.

We show that an arbitrary classification can be optimized in two aspects. The first is to introduce coherence in the decision, while the second is to develop a fully-automatic solution, saving time and human-resources. In a first ideal approach, an automatic solution is presented. Then, we introduce coherence to the previous solution.

The main benefits of Multitree Committee are its adaptability to an arbitrary number of sensors n, classification range $[r]$, and level of inspection, without a need of misclassification probabilities (whose estimation is non-trivial) and historical feedback, since the new classification can extend the training set.

Regarding these elements and the recommendations from the port operators, as a future work we want to contrast stochastic-based mechanisms with respect to the current development, and tune our algorithm here described adding randomness, in order to cope with purely deterministic attacks.

8. Acknowledgments

This work has been partially supported by Administración Nacional de Puertos (ANP) and Dirección Nacional de Aduanas (DNA), from Montevideo, Uruguay. We express our most sincere gratitude to Dr. Franco Robledo for his support and effort in order to set-up the contract between Facultad de Ingeniería and Administración Nacional de Puertos.

References

[1] Anand, S., Madigan, D., Mammone, R., Pathak, S., & Roberts, F. (2006). Experimental analysis of sequential decision making algorithms for port of entry inspection procedures. In *ISI Lecture Notes in Computer Science* (pp. 319–330). Springer-Verlag.

[2] Boros, E., Fedzhora, L., Kantor, P., Saeger, K., & Stroud., P. (2009). A large-scale linear programming model for finding optimal container inspection strategies. *Naval Research Logistics (NRL)*, *56*, 404–420. URL: `http://dx.doi.org/10.1002/nav.20349 Last Access: 23 June 2013.` doi:10.1002/nav.20349.

[3] Bryant, R. E. (1992). Symbolic boolean manipulation with ordered binary-decision diagrams. *ACM Computing Surveys*, *24*, 293–318.

[4] Candy, J. (2013). Sequential threat detection for harbor defense: An x-ray physics-based bayesian approach. In *OCEANS - Bergen, 2013 MTS/IEEE* (pp. 1–9). doi:10.1109/OCEANS-Bergen.2013.6607948.

[5] Concho, A. L., & Ramirez, J. (2010). An evolutionary algorithm for port-of-entry security optimization considering sensor thresholds. *Reliability Engineering & System Safety*, *95*, 255–266.

[6] Dedekind, R. (1897). Uber Zerlegungen von Zahlen durch ihre grossten (Decomposition of boolean functions). *Werke*, *II*, 103–148.

[7] Dreiding, R. A., & McLay, L. A. (2013). An integrated model for screening cargo containers. *European Journal of Operational Research*, *230*, 181–189.

[8] Elsayed, E. A., Young, C. M., Xie, M., Zhang, H., & Zhu, Y. (2009). Port-of-entry inspection: Sensor deployment policy optimization. *IEEE T. Automation Science and Engineering*, *6*, 265–276.

[9] Fu, Z., & Mae, F. (2001). A computational study of using genetic algorithms to develop intelligent decision trees. In *Evolutionary Computation, 2001. Proceedings of the 2001 Congress on* (pp. 1382–1387 vol. 2). volume 2. doi:10.1109/CEC.2001.934352.

[10] Graneri, J., Moscatelli, S., Romero, P., Tansini, L., & Viera, O. (2015). Monte-carlo mirror algorithm for the port-of-entry inspection problem. *International Journal in Operations Research*, *To appear*.

[11] Hyafil, L., & Rivest, R. (1976). Constructing Optimal Binary Decision Trees is NP-complete. *Information Processing Letters*, *5*, 15–17.

[12] Li, J. P., Balazs, M. E., Parks, G. T., & Clarkson, P. J. (2003). Erratum: A species conserving genetic algorithm for multimodal function optimization. *Evolutionary Computation*, *11*, 107–109.

[13] Madigan, D., Mittal, S., & Roberts, F. (2007). Sequential decision making algorithms for port of entry inspection: Overcoming computational challenges. In *ISI* (pp. 1–7).

[14] McLay, L. A., & Dreiding, R. A. (2012). Multilevel, threshold-based policies for cargo container security screening systems. *European Journal of Operational Research*, *220*, 522–529.

[15] Ramirez, J. (2008). Port-of-entry safety via the reliability optimization of container inspection strategy through an evolutionary approach. *Reliability Engineering and System Safety*, *93*, 1698–1709.

[16] Stroud, P. D. (2003). Enumeration of increasing boolean expressions and alternative digraph implementations for diagnostic. In *Applications, Proceedings Volume IV, Computer, Communication and Control Technologies* (pp. 328–333). Springer-Verlag.

[17] Wein, L., Wilkins, A., Baveja, M., & Flynn, S. (2006). Preventing the importation of illicit nuclear materials in shipping containers. *Risk Anal*, *26*, 1377–93.

[18] Wiedemann, D. (1991). A computation of the eighth dedekind number. *Order*, *8*, 5–6. URL: `http://dx.doi.org/10.1007/BF00385808 Last Access: 2 July 2013`. doi:10.1007/BF00385808.

[19] Xue, L., & Villalobos, J. (2012). A multi-objective optimization primary planning model for a poe (port-of-entry) inspection. *Journal of Transportation Security*, *5*, 217–237. URL: `http://dx.doi.org/10.1007/s12198-012-0093-8. Last access: 2 July 2013`. doi:10.1007/s12198-012-0093-8 Last Access: 2 July 2013.

[20] Young, C. M. (2010). *Modeling and optimization of container inspection systems*. Ph.D. thesis Rutgers State University of New Jersey New Jersey, USA. URL: `http://rucore.libraries.rutgers.edu/rutgers-lib/30503/` `Last Access: 23 August 2013`.

[21] Young, C. M., Li, M., Zhu, Y., Xie, M., Elsayed, E. A., & Asamov, T. (2010). Multiobjective optimization of a port-of-entry inspection policy. *IEEE T. Automation Science and Engineering*, *7*, 392–400.