

# **Succinct Specifications of Portable Document Access Policies**

**Marina Bykova, Mikhail Atallah**

CERIAS and Department of Computer Sciences  
Purdue University

ACM Symposium on Access Control Models and Technologies  
(SACMAT'04)  
June 2004

# Outline

- Problem description
- Probabilistic model
- Deterministic model
- Implementation notes
- Conclusions

## Problem Description

- The model
  - We are given a very large data repository
  - Access is payment-based
  - Each customer can request a subscription to any subset of items
- Becomes important as the number and the level of maturity of on-line document collections grow
- Might not be challenging to solve without additional constraints

## Problem Description (cont.)

- Constraints
  - For customer privacy subscriptions are not stored at the server
  - Limited-capacity storage devices are used for policy configurations
    - important in the case of smart cards
    - results in inability to precisely represent all subsets
    - introduces “false positives”
- The goal: minimal cost
  - The cost associated with “false positives” should be as small as possible

## Problem Description (cont.)

- Two models
  1. Deterministic model
    - all customers and their orders are known in advance
  2. Probabilistic model
    - no order information is known before policy assignment
    - each document has a probability of being chosen by a single subscriber
- Two optimization types
  1. Minimizing *total* cost of false positives over all customers
  2. Minimizing *maximum* cost of false positives for a single subscription

## Notation

- Repository contains  $n$  elements  $1, \dots, n$ .
- Access to document  $i$  can be purchased at the price  $c_i$
- Binary strings  $m$  bits long ( $m < n$ ) are used to represent access rights
- Every subscription bitstring is constructed using bitwise OR of the bitstrings of the documents composing the order
- The “ $\geq$ ” operation on access rights is defined as a bitwise  $\geq$  comparison of two bitstrings

## Probabilistic Model

- Each document  $i$  has access probability  $0 < p_i \leq 1$
- All probabilities  $p_1, \dots, p_n$  are independent
- The “cost” of a policy assignment is now a sum of probabilities of all subsets of the documents, with each subset weighted by the costs of the false positives in it

## Probabilistic Model (cont.)

- Minimizing the total cost of false positives
  - Trying all bitstrings for each document, for all possible document subsets, is impractical
  - Empirical observation: setting only one bit to 1 in an access bitstring corresponding to a document approximates the optimum solution rather well
  - Still does not allow for an efficient solution
    - the problem is NP-hard
    - reduction from partitioning of  $n$  items into  $m$  buckets such that the sum of the squares of bucket weights is below a threshold

## Minimizing the Total Cost in Probabilistic Model

- One bit per document
  - The goal can be achieved by partitioning  $n$  documents into  $m$  groups
  - “Cost”  $C_i$  of a group  $i$  is:

$$C_i = \sum_{j=1}^{s_i} c_{ij}(1 - p_{ij}) - \left( \sum_{j=1}^{s_i} c_{ij} \right) \left( \prod_{j=1}^{s_i} (1 - p_{ij}) \right)$$

- The total cost is the sum of groups’ costs
- Given a policy assignment, the cost is computed in linear time
- We give an efficient algorithm for cases when all  $c_i$ ’s are equal (e.g.,  $c_i = 1$ )

## Minimizing the Total Cost in Probabilistic Model (cont.)

- One bit per document — Solution

- $s_i$  denotes the size of group  $i$
- “Cost” of group  $i$  is

$$C_i = \sum_{j=1}^{s_i} (1 - p_{ij}) - s_i \prod_{j=1}^{s_i} (1 - p_{ij})$$

- Contiguous grouping of  $n$  sorted items into  $m$  groups gives optimal results
- Dynamic programming algorithm gives a solution in  $O(mn^2)$  time

## Minimizing the Total Cost in Probabilistic Model (cont.)

- One bit per document, one document at a time
  - Each customer includes only one document in an order
  - The sum of document probabilities  $p_i$ 's is now  $\leq 1$
  - The group cost becomes  $C_i = \sum_{j=1}^{s_i} p_{i_j} \sum_{\substack{k=1 \\ k \neq j}}^{s_i} c_{i_k}$
  - Similarly, when all  $c_i$ 's are equal, a dynamic programming algorithm solves the problem in  $O(mn^2)$  time
    - group cost is  $C_i = (s_i - 1) \sum_{j=1}^{s_i} p_{i_j}$
    - “monotonicity”: a group composed of documents with larger probabilities has smaller size
    - dynamic programming approach tests all choices for partitioning in  $O(mn^2)$  time

## Minimizing the Maximum Cost in Probabilistic Model

- Individual subscription order is considered, any set of documents is possible
- When all document costs  $c_i$  are equal, simply partition  $n$  documents into  $m$  groups of  $n/m$  documents each
- When  $c_i \neq c_j$ , we need to minimize

$$C = \sum_{i=1}^n c_i - \sum_{j=1}^m \min_{i=1 \text{ to } n} \{c_i \mid i \in S_j\}$$

where  $S_i$  is the document set of group  $i$

- Optimal partitioning can be done in  $O(n)$  time
- Total algorithm runs in  $O(n \log n)$  time

## Deterministic Model

- There are  $k$  subscribers  $1, \dots, k$
- Subscriber  $i$  requests  $s_i$  documents  $i_1, \dots, i_{s_i}$
- Optimal solution to the *total* cost of false positives problem requires

$$C = \min \left\{ \sum_{i=1}^k C_i \right\} = \min \left\{ \sum_{i=1}^k \left( f^{-1} \left( \bigvee_{j=1}^{s_i} r_{i_j} \right) - \sum_{j=1}^{s_i} c_{i_j} \right) \right\}$$

- Optimal solution to the *maximum* cost of false positives problem is computed as

$$C = \min \left\{ \max_{i=1 \text{ to } k} C_i \right\} = \min \left\{ \max_{i=1 \text{ to } k} \left( f^{-1} \left( \bigvee_{j=1}^{s_i} r_{i_j} \right) - \sum_{j=1}^{s_i} c_{i_j} \right) \right\}$$

where  $f^{-1}(r)$  computes the cost of policy  $r$

## Deterministic Model (cont.)

- Any general solution to cost minimization is intractable
  - the problem is NP-hard
  - reduction from the graph bisection problem
- Practical heuristic: use probabilistic approach to solve deterministic (compute probabilities, etc.)

## Implementation Issues

- Static policy assignment makes sharing of information about false positives possible
  - the framework is best suited for periodic subscriptions with policy refreshment
  - performance can “drift” from optimality between policy re-generations
  - a “ $t$  strikes and you are out” strategy can be employed
- Document probabilities can be refined over time
- Randomization can be introduced into the policy assignment process

## Conclusions and Future Work

- This work explores the problem of policy assignment optimization under space constraints
- Efficient algorithms are developed for some settings, while others are shown to be intractable
- Future directions include:
  - considering dependency between documents
  - allowing for different types of documents
  - exploring the problem for structured sets of documents
- These slides are available at <http://www.cs.purdue.edu/homes/mbykova/papers/sacmat04-slides.{ps,pdf}>