Tracing pirate cards as part of the satellite video broadcasting

Ulrich Aivodji, Alexandre Gonzalvez, Pascal Lefevre, Brandon Dravie

Supervisors: Cécile Delerablée, Thomas Baignères

REDOCS’16 Report

28th October 2016
1. Problem

2. Performance Metrics

3. Strategy 1

4. Strategy 2

5. Strategy 3

6. Bilan & Perspective
<table>
<thead>
<tr>
<th></th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Problem</td>
</tr>
<tr>
<td>2</td>
<td>Performance Metrics</td>
</tr>
<tr>
<td>3</td>
<td>Strategy 1</td>
</tr>
<tr>
<td>4</td>
<td>Strategy 2</td>
</tr>
<tr>
<td>5</td>
<td>Strategy 3</td>
</tr>
<tr>
<td>6</td>
<td>Bilan &amp; Perspective</td>
</tr>
</tbody>
</table>
Problem

Satellite broadcasting
Problem

Hacking
Problem

Provider:
Pirate client
Problem

Provider:
Pirate client

Killer ECM
Problem

Provider: Pirate client

Killer ECM

Id target
Pirate Strategies

1. $\text{strategy1}(r, CW_0, \ldots, CW_{n-1}) \rightarrow CW_0$

2. $\text{strategy2}(r, CW_0, \ldots, CW_{n-1}) \rightarrow \begin{cases} 
\text{majority}(CW_0, \ldots, CW_{n-1}) & \text{if } n \text{ is odd} \\
CW_0 & \text{else}
\end{cases}$

3. $\text{strategy3}(r, CW_0, \ldots, CW_{n-1}) \rightarrow CW_{r \mod n}$
Table of contents

1 Problem

2 Performance Metrics

3 Strategy 1

4 Strategy 2

5 Strategy 3

6 Bilan & Perspective
Metrics

- CPU Time \((s)\)
- Collateral damage \(ColD\) (Avg, stddev): \(\sum(1 - \frac{id_i}{N})\)
- QoS of the pirate (Avg): \(100 \times \frac{t}{T}\) (\(T\): number of normal ECMs, \(t\): number of correct \(cw\))
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Problem</td>
</tr>
<tr>
<td>2</td>
<td>Performance Metrics</td>
</tr>
<tr>
<td>3</td>
<td>Strategy 1</td>
</tr>
<tr>
<td>4</td>
<td>Strategy 2</td>
</tr>
<tr>
<td>5</td>
<td>Strategy 3</td>
</tr>
<tr>
<td>6</td>
<td>Bilan &amp; Perspective</td>
</tr>
</tbody>
</table>
General principle

- Hypothesis: The traitor always uses the same card.
- Goal: Locate the card by using a minimum number of tracking ECM. Killing the card will
- Solution: Binary search (average number of iterations $\log(n)$)
Benchmark

<table>
<thead>
<tr>
<th>Method</th>
<th>CPU Time (s)</th>
<th>Collateral damage</th>
<th>QoS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Avg</td>
<td>Stddev</td>
<td>Avg</td>
</tr>
<tr>
<td>Binary Search</td>
<td>54.37</td>
<td>14.11</td>
<td>8.14</td>
</tr>
<tr>
<td>Ternary Search</td>
<td>54.18</td>
<td>17.52</td>
<td>10.78</td>
</tr>
</tbody>
</table>

Table: Benchmark for 100 runs and nbCard = 10
Table of contents

1 Problem

2 Performance Metrics

3 Strategy 1

4 Strategy 2

5 Strategy 3

6 Bilan & Perspective
Strategy 2

Notations

- $M_t$: pirate response to ECM tracer $t$
- $M_t \in L$: majority of pirate cards identifiers are $< t$ (on the left side)
- $M_t \in R$: majority of pirate cards identifiers are $\geq t$ (on the right side)
Strategy 2: Why binary search works.

Proposition

Let $p = 2k + 1$ cards (majority vote) and $t' < t$ two tracers ECM. $M_t \in L$ and $M_{t'} \in R \implies \exists \text{Id}_p \in [t', t]$. 

![Diagram](image-url)
Algorithm

Pivots

- \( p \leftarrow 0 \)
- \( p' \leftarrow N - 1 \)

Details

- Stops when \( |p - p'| = 1 \)
- ECM tracer \( t_m \leftarrow \lfloor (p + p') / 2 \rfloor \)
  
  if \( M_{t_m} \in L \), \( p' \leftarrow t_m \)
  
  else \( p \leftarrow t_m \)
## Benchmark Strategy II

<table>
<thead>
<tr>
<th></th>
<th>CPU Time (s)</th>
<th>Collateral damage</th>
<th>QoS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optimal approach</td>
<td>249.82</td>
<td>50.35</td>
<td>17.24</td>
</tr>
<tr>
<td>Paper approach [Tas05]</td>
<td>94.69</td>
<td>50.55</td>
<td>15.10</td>
</tr>
</tbody>
</table>

**Table:** Benchmark for 100 runs and nbCard = 10

---

Strategy 3

Description
Strategy 3
Description

Pirate Strategy

Own a number $n$ of cards, and generates randomly and uniformly an number $r$ larger than $n$. 
**Strategy 3**

Description

**Pirate Strategy**

Own a number $n$ of cards, and generates randomly and uniformly an number $r$ larger than $n$.

- made only of correct values $cw$
Strategy 3

Description

Pirate Strategy

Own a number \( n \) of cards, and generates randomly and uniformly an number \( r \) larger than \( n \).

- made only of correct values \( cw \)
- made only of incorrect values of \( cw \)
Strategy 3

Description

Pirate Strategy

Own a number $n$ of cards, and generates randomly and uniformly an number $r$ larger than $n$.

- made only of correct values $cw$
- made only of incorrect values of $cw$
- made both of correct and incorrect values of $cw$
Strategy 3

Algorithm

Population : N-1 cards
Strategy 3

Algorithm

\[
\frac{(P-1) \times N}{P}
\]

P pirates cards → P - 1 intervals
Strategy 3

Algorithm

Condition for dichotomy

In $[A;B]$, if $NbrCardsFalse > 0$, then at least a pirate card is present.
Strategy 3

Algorithm

\[
0 \quad \frac{(P-m) \times N}{P} \quad N - 1
\]

With \( m \) intervals.
Strategy 3

Algorithm

\[
\frac{(P - m) \times N}{P} \quad \quad \quad \quad \quad N - 1
\]

**Condition for dichotomy**

In \([A; B]\), if \(NbreCardsFalse - (nbCardsMute + 0.6) \times Cst > 0\), then at least a pirate card is present.
## Benchmark Strategy III

<table>
<thead>
<tr>
<th>Metric</th>
<th>Heuristic approach</th>
<th>CPU Time (s)</th>
<th>Collateral damage</th>
<th>QoS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Avg</td>
<td>Stddev</td>
<td>Avg</td>
</tr>
<tr>
<td>Average</td>
<td>631.29</td>
<td>68865</td>
<td>35630</td>
<td>57</td>
</tr>
</tbody>
</table>

**Table**: Benchmark for 100 runs and nbCard = 10
General principle for Strategy III version II

- Return a set of small intervals that have a good probability to contain id of the traitor’s cards.
- Let $S = \{0, 1, 2, \ldots, n-1\}$ be the set of all the cards (regular user and traitors).
- Divide $S$ in 100 subsets and select subsets $S_j$ that pass the test.
- The test take a subset $S_j = [a, b]$, uses $a$ and $b$ as input for a tracking ECM send $nbSample$ times.
- Let $ProbA$ be the chance to have negative response with tracking ECM $a$ and $ProbB$ be the chance to have positive response with tracking ECM $b$.
- Reject $S_j$ if $\text{abs}(ProbA - ProbB) > \epsilon$ and accept $S_j$ otherwise.
- Repeat until $|S_j| \leq 1000$. 
Table of contents

1 Problem
2 Performance Metrics
3 Strategy 1
4 Strategy 2
5 Strategy 3
6 Bilan & Perspective
Bilan

- Optimal counter attack against strategy I and II
- 2 heuristic approaches for strategy III
Perspectives

- Find theoretical bound for strategy III
- Explore game theory alternative (Bilevel optimization)

\[
\min_{id_i} \sum_i (1 - \frac{id_i}{N}) \\
\text{s.t. } QoS(pirate) \leq \epsilon \\
id_i \in \{0, 1\}
\]
## Bonus

### Content

- Challenges
- Knowledge

### Interesting tools and methods

- Teamwork with efficiency (AGILE method)
- Tools: GitHub, CollabEdit

### Social

- Contacts, colleagues, friends, fun...