Optimal Patrol Planning for Green Security Games with Black-Box Attackers

Haifeng Xu\textsuperscript{1}, Ben Ford\textsuperscript{1}, Fei Fang\textsuperscript{2}, Bistra Dilkina\textsuperscript{1}, Andrew Plumtre\textsuperscript{3}, Milind Tambe\textsuperscript{1}, Margaret Driciru\textsuperscript{4}, Fred Wanyama\textsuperscript{4}, Aggrey Rwetsiba\textsuperscript{4}, Mustapha Nsubaga\textsuperscript{4}, Joshua Mabonga\textsuperscript{4}

\textsuperscript{1}University of Southern California (USC)  
\textsuperscript{2}Carnegie Mellon University (CMU)  
\textsuperscript{3}Wildlife Conservation Society  
\textsuperscript{4}Uganda Wildlife Authority
The Classical Stackelberg Security Game Paradigm

- Stackelberg Game
  - Defender (leader): use limited resources to protect critical targets
  - Attacker (follower): long-term surveillance, well-planned (thus perfectly rational)

Flights  Ferries  Airports  Road Networks
A Rapidly Growing Trend: Green Security Domains

Endangered Wildlife

Today
≈ 3,200

100 years ago
≈ 60,000

Fisheries

Environmental Resources
Challenges for Patrol Planning in Green Security Games

- Attacker’s bounded rationality $\Rightarrow$ intricate attacker (behavior) models

**Challenge 1:**

How to optimize patrolling against these complicated attacker models?

Do we have to design a different algorithm for each attacker model?
Challenges for Patrol Planning in Green Security Games

- Attackers may have partial real-time surveillance
  - Can observe rangers’ current move and infer where they go next

“Those (poachers) would simply observe the rangers and base their offending patterns on the schedules of the rangers”

Challenge 2:
How to deal with attacker’s (partial) real-time surveillance?
Our Contributions:

- A new patrol planning framework OPERA (Optimal patrol Planning with Enhanced RA Randomness)
  - Work for any attacker model (under mild assumptions)
  - Mitigate negative effects of attacker’s real-time surveillance with enhanced randomness

- Test performances on real-world data from Uganda
Outline

- Motivation and Game Model
- Optimal Patrol Planning Against Black-Box Attackers
- Experimental Evaluation
Outline

- Motivation and Game Model
- Optimal Patrol Planning Against Black-Box Attackers
- Experimental Evaluation
Motivation Domain: Wildlife Protection in Uganda

**Forest Area**: QEPA
- Covers 2520 sq. km
- Divided into grids of 1km × 1km

**Poachers**: set trapping tools (e.g., snare)

**Rangers**: conduct patrols

**Our Goal**: maximize catches of snares

**Collaborators**: Wildlife Conservation Society, Uganda Wildlife Authority,
Motivation Domain: Wildlife Protection in Uganda

Patrol post
(one patroller)
Defender Strategy

**Claim:** a mixed strategy $\iff$ one-unit fractional flow from $v_{11}$ to $v_{1T}$

**Observe:** a pure strategy = a path from $v_{11}$ to $v_{1T}$

**Def:** patrol effort at cell $i$ = the aggregated flow through cell $i$

Time-unrolled Graph
Outline

- Motivation and Game Model
- Optimal Patrol Planning Against Black-Box Attackers
- Experimental Evaluation
The Single-Step Planning Task

Timeline:

Goal: maximize catches of snares against any given attacker model
- Attacker model: (current patrolling effort + other features) \(\rightarrow\) predicted snare presence
But... Many Complicated Attacker Models

Graphical Model [Nguyen et al.'16]
But...Many Complicated Attacker Models

Decision Trees [Kar et al.’ 17]
But…Many Complicated Attacker Models

Markov Random Field [Gholami et al.’17]
But... Many Complicated Attacker Models

More are coming...

Deep Neural Networks ???
How to optimize over these complicated attacker models?
Our Idea: Treat It as a Black-Box Function

For each cell $i$:

- Current patrol effort at $i$
- Terrain features
- Animal density
- Previous effort at $i$

......

$\rightarrow$ Attacker Model

$\rightarrow$ Prob. of detecting a snare at $i$ in current period
Our Idea: Treat It as a Black-Box Function

For each cell $i$:

Assumption: $g_i$ depends discretely on the current patrol effort

- Patrol levels in $\{0, 1, 2, \ldots, m\}$
  - **Thresholds** to classify patrol efforts into levels
- $g_i(0), g_i(1), \ldots, g_i(m)$ are the predicted probabilities for each level
- A good approximation when $g_i$ is Lipchitz continuous in effort and $m$ sufficiently large
The Optimization Task

Design patrol levels $l_1, \ldots, l_m$ (induced by patrol efforts) to

$$\text{maximize } \sum_{i=1}^{N} g_i(l_i)$$

- Main Challenge: black-box representation results in combinatorial decision making problem under constraints
**Theorem:** Computing optimal mixed strategy is NP-hard.

Idea: reduction from Knapsack Problem

- $m$ patrol levels with thresholds: $\alpha_0 < \alpha_1, ..., < \alpha_m$
- $g_i(i) = p_i$ and $g_i(j) = 0, \forall j \neq i$

Goal: with 1 unit patrol budget, decide for each $i$ to patrol with $\alpha_i$ (reward $p_i$) or patrol with 0 (reward 0)

$\iff$

Packing $m$ items (weight $\alpha_i$, value $p_i$) to a 1 unit bag
Our Solution

A compact *mixed integer linear program* formulation for the optimization problem

\[
\begin{align*}
\text{maximize} & \quad \sum_{i=1}^{N} \left( g_i(0) + \sum_{j=1}^{m} z_i^j \cdot [g_i(j) - g_i(j - 1)] \right) \\
\text{subject to} & \quad x_i \geq \sum_{j=1}^{m} z_i^j \cdot [\alpha_j - \alpha_{j-1}], \\
& \quad x_i \leq \alpha_1 + \sum_{j=1}^{m} z_i^j \cdot [\alpha_{j+1} - \alpha_j], \\
& \quad z_i^1 \geq z_i^2 \geq \ldots \geq z_i^m, \\
& \quad z_i^j \in \{0, 1\}, \\
& \quad x_i = \sum_{t=1}^{T} \left[ \sum_{e \in \sigma^+(v_t,i)} f(e) \right], \\
& \quad \sum_{e \in \sigma^+(v_t,i)} f(e) = \sum_{e \in \sigma^-(v_t,i)} f(e), \\
& \quad \sum_{e \in \sigma^+(v_{T,1})} f(e) = \sum_{e \in \sigma^-(v_{1,1})} f(e) = 1 \\
& \quad 0 \leq x_i \leq 1, \quad 0 \leq f(e) \leq 1,
\end{align*}
\]
Our Solution

A compact *mixed integer linear program* formulation for the optimization problem

- Involve a particular technique to linearize the problem
- Scalable to problems with, e.g., 100 targets and 5 patrol levels

**However**

- Output a mixed strategy randomizing over only a few paths
- Unavoidable – efficient solvers are *designed to* find small-support solutions
- Vulnerable to attacker’s (partial) real-time surveillance
Add Extra Randomness by Entropy Maximization

- Many mixed strategies implement the same patrolling effort

- We compute the one that maximizes (Shannon) entropy
  - Usually support on a much larger set of paths
  - Difficult to learn

- There is an efficient algorithm to compute max-entropy distribution here
  - Convex analysis, combinatorial optimization, duality theory
Extension: Multi-Step Planning

Timeline:

Previous period  Current period  Next period

\[ g_i^1(l_i^1) \quad g_i^2(l_i^1, l_i^2) \]

Goal: maximize aggregated total catch

\[ \text{maximize} \quad \sum_{i=1}^{N} g_i^2(l_i^2, l_i^1) + \sum_{i=1}^{N} g_i^1(l_i^1) \]
Outline

- Motivation and Game Model
- Optimal Patrol Planning Against Black-Box Attackers
- Experimental Evaluation
Real-World Data Set from QEPA

Rangers record captures of snares
- From 2003 – 2017
- 39 patrol posts
- We test on post 11, 19, 24 (the mostly attacked)

Collaborators: Wildlife Conservation Society, Uganda Wildlife Authority,
Experiment 1: Compare with Baseline Algorithms

**OPERA:**
- bagging ensemble model [Gholami et al.’17] (two levels: low and high)

**OPP:** Optimal Patrol Planning

Another Two Baselines
- **GREED:** greedily pick the next reachable cell to patrol
- **RAND:** randomly pick the next reachable cell to patrol
**Experiment 1: Compare with Baseline Algorithms**

<table>
<thead>
<tr>
<th></th>
<th>#Detection</th>
<th>#Cover</th>
<th>#Routes</th>
<th>Entropy</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPERA</td>
<td>15/19</td>
<td>20/47</td>
<td>61</td>
<td>4.0</td>
</tr>
<tr>
<td>OPP</td>
<td>15/19</td>
<td>20/47</td>
<td>10</td>
<td>2.0</td>
</tr>
<tr>
<td>GREED</td>
<td>5/19</td>
<td>4/47</td>
<td>84</td>
<td>4.4</td>
</tr>
<tr>
<td>RAND</td>
<td>4/19</td>
<td>6/47</td>
<td>89</td>
<td>4.5</td>
</tr>
</tbody>
</table>

Comparisons of Different Criteria for Patrol Post 11

- **#Detection**: \(a/b\) → out of \(b\) predicted attacks, the algorithm detects \(a\) attacks
- **#Cover**: \(a/b\) → out of \(b\) cells, \(a\) of them are covered with *high*
Experiment 1: Compare with Baseline Algorithms

<table>
<thead>
<tr>
<th></th>
<th>#Detection</th>
<th>#Cover</th>
<th>#Routes</th>
<th>Entropy</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPERA</td>
<td>6/6</td>
<td>24/72</td>
<td>22</td>
<td>2.6</td>
</tr>
<tr>
<td>OPP</td>
<td>6/6</td>
<td>24/72</td>
<td>6</td>
<td>1.3</td>
</tr>
<tr>
<td>GREED</td>
<td>2/6</td>
<td>2/72</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>RAND</td>
<td>2/6</td>
<td>6/72</td>
<td>90</td>
<td>4.5</td>
</tr>
</tbody>
</table>

Comparisons of Different Criteria for Patrol Post 19


## Experiment 2: Compare with Past (Real) Patrolling

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Post 11 OPERA</th>
<th>Post 11 Past</th>
<th>Post 19 OPERA</th>
<th>Post 19 Past</th>
<th>Post 24 OPERA</th>
<th>Post 24 Past</th>
</tr>
</thead>
<tbody>
<tr>
<td>#Detections</td>
<td>15/19</td>
<td>4/19</td>
<td>6/6</td>
<td>5/6</td>
<td>4/4</td>
<td>3/4</td>
</tr>
<tr>
<td>#Cover</td>
<td>20/47</td>
<td>6/47</td>
<td>24/72</td>
<td>11/72</td>
<td>20/59</td>
<td>14/59</td>
</tr>
</tbody>
</table>
Take-Away Message

- An efficient patrol planning tool that
  - Optimize against very general class of attacker models
  - Mitigate attacker real-time surveillance by adding extra randomness

Special Thanks to Wildlife Conservation Society, Uganda Wildlife Authority

Thank You