Part II: Game theory for crime prevention

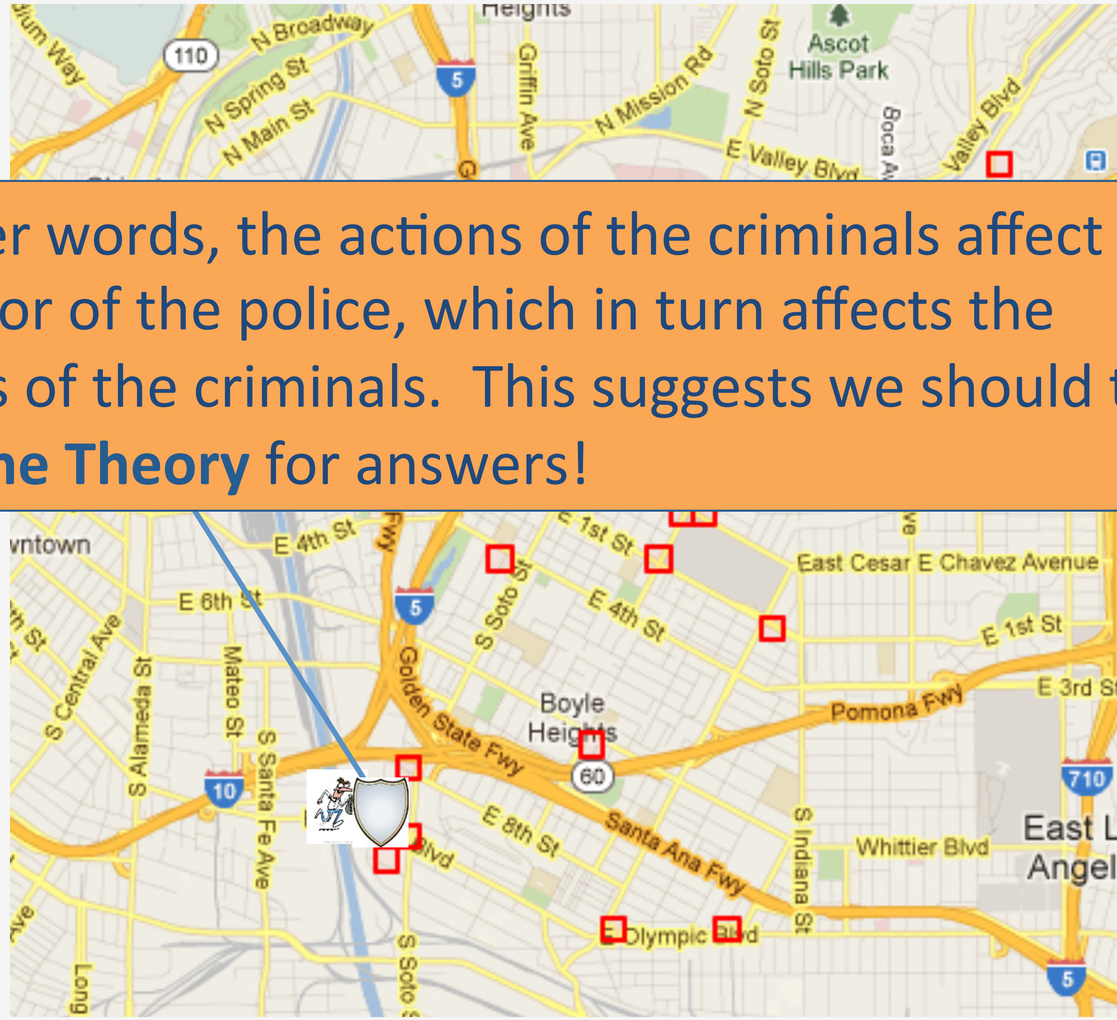
If we want to actually *use* crime predictions, there is a chicken and egg type problem…

Attackers pr
like these lo

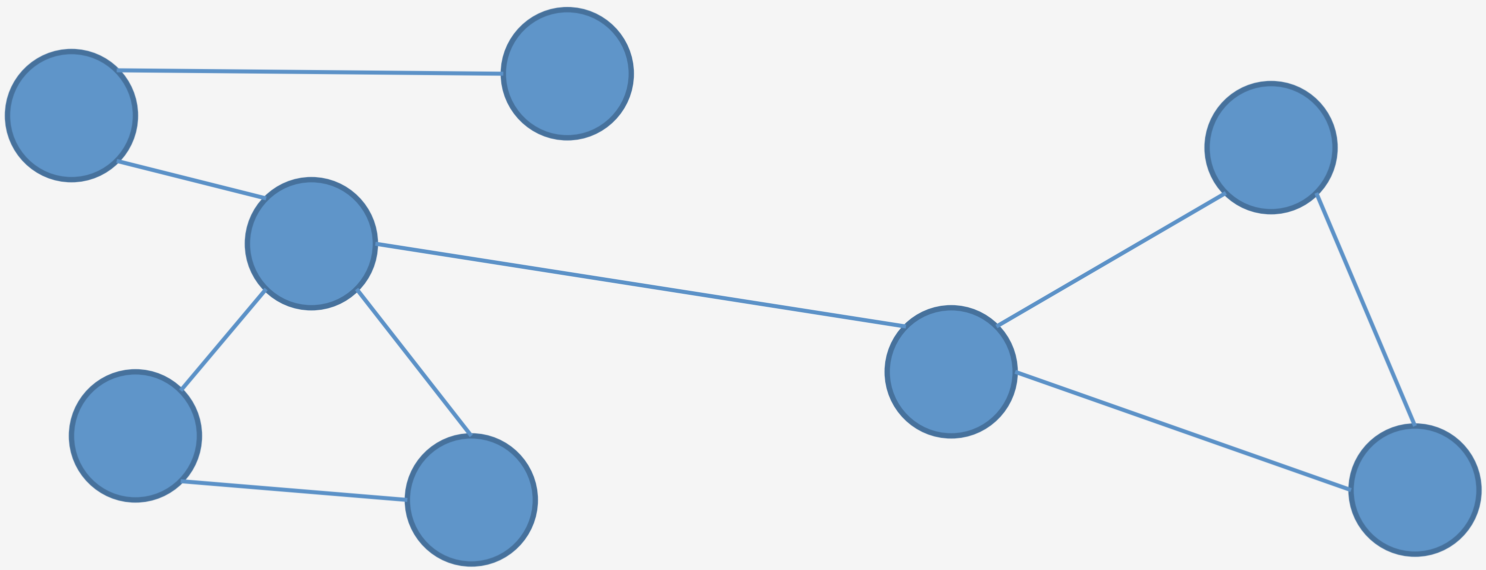
In other words, the actions of the criminals affect the behavior of the police, which in turn affects the actions of the criminals. This suggests we should turn to **Game Theory** for answers!

But if we s
patrols the…

criminals may
just go somewhere
else.

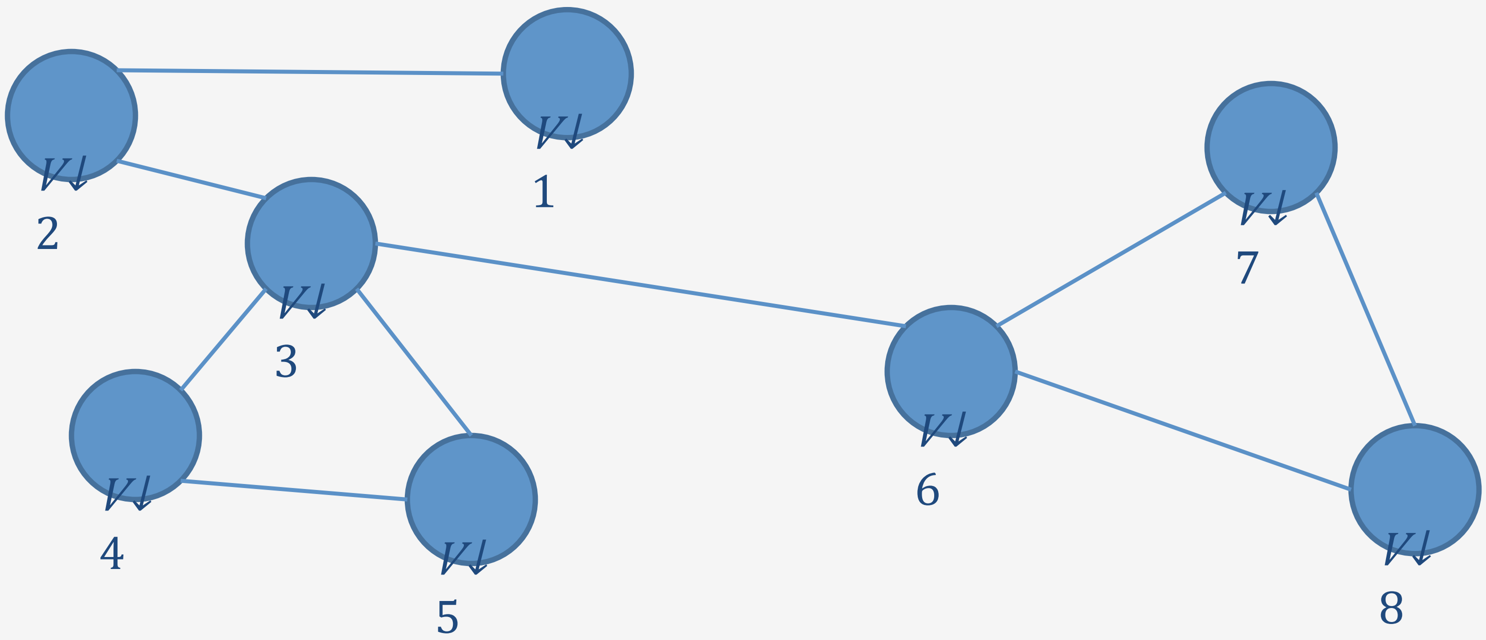


There is a well-studied framework for such scenarios: Security Games (SG)



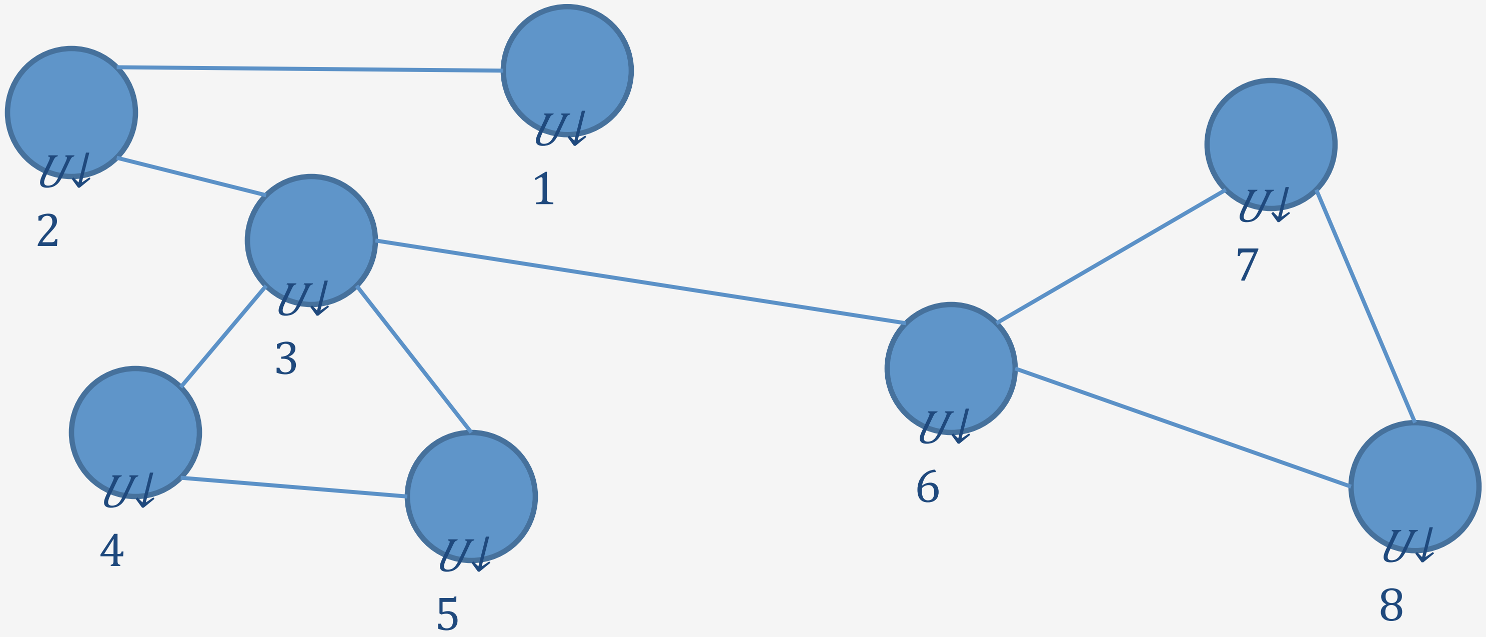
Suppose there is a set of *N* locations (perhaps nodes on a graph) that an attacker might like to attack (burgle, steal cars, set off a bomb, etc).

There is a well-studied framework for such scenarios: Security Games (SG)



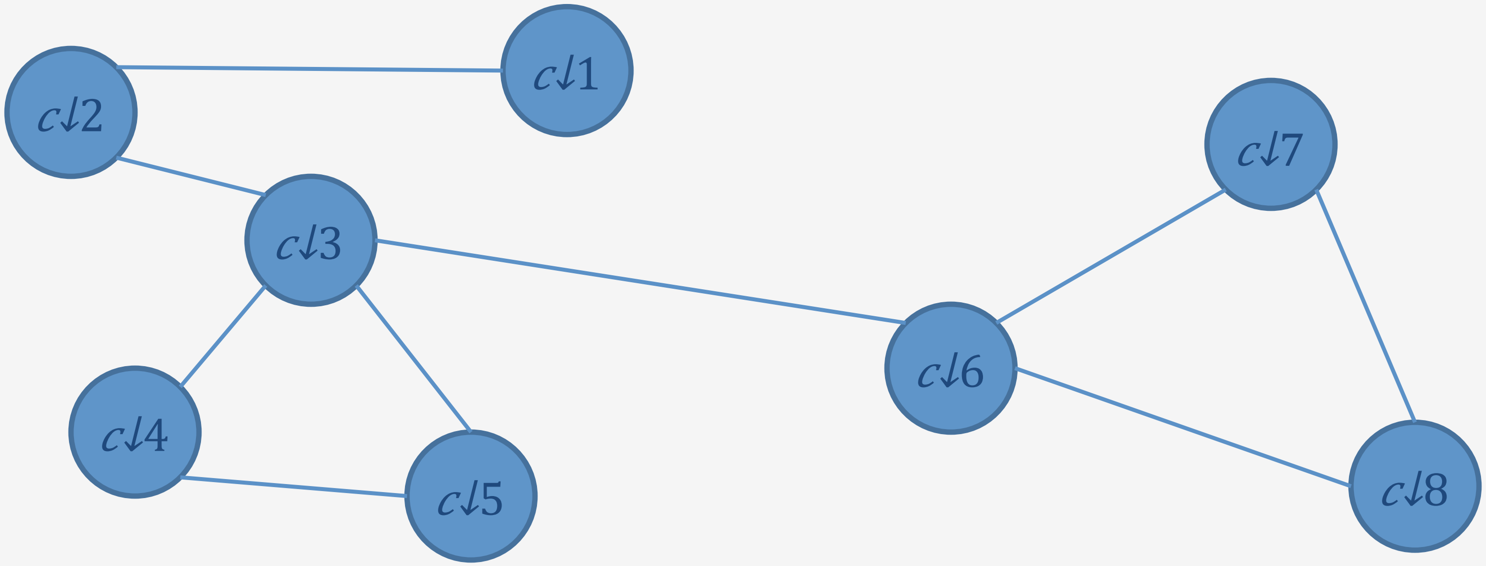
But, even if success were guaranteed, the locations are not all of equal value $V_i \geq 0$ to the attacker (more nice houses to burgle, higher value targets).

There is a well-studied framework for such scenarios: Security Games (SG)



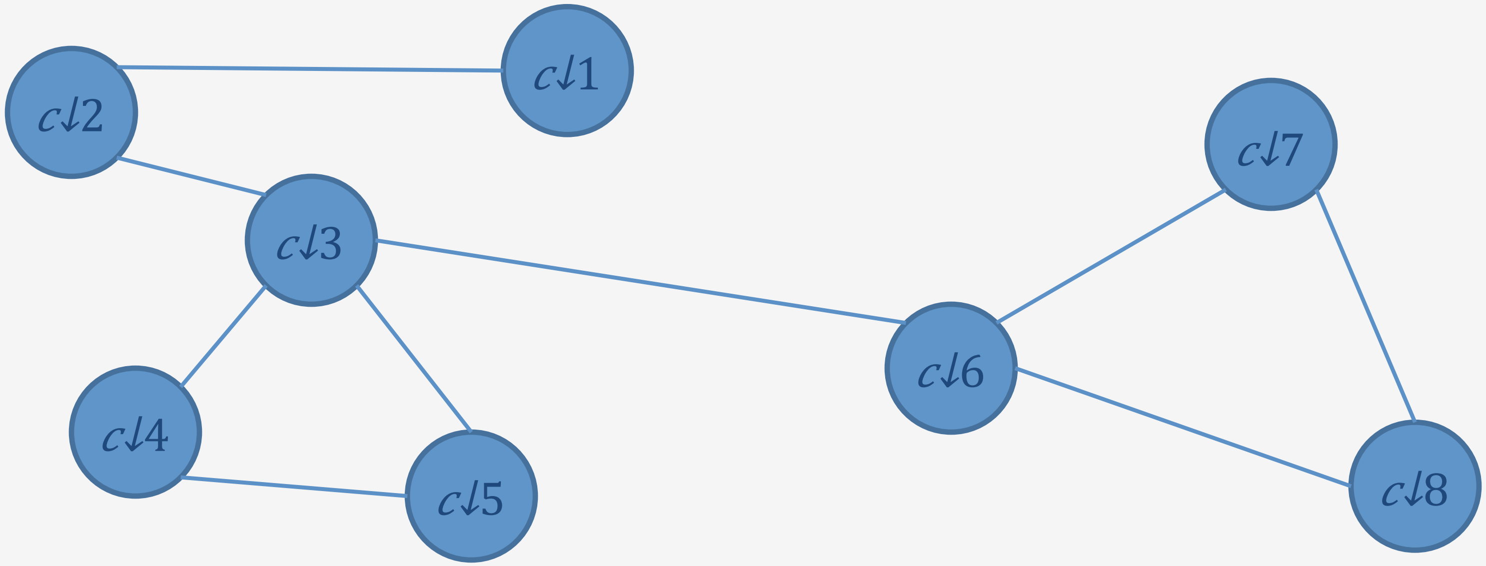
Now suppose a defender is trying to defend the targets. In defender presence, the attacker is foiled, and in defender absence the attacker is successful. Defender has utilities $U_i \leq 0$ for successful attacks.

There is a well-studied framework for such scenarios: Security Games (SG)



So, the defender must choose a defense strategy in the form of a schedule of resources, yielding a coverage vector c. The component c_i is the probability location i will be defended in any given “play” of the game.

There is a well-studied framework for such scenarios: Security Games (SG)



Key Point: It is assumed that attackers always do worse at a location if a defender is present, while defenders always do better.

Take a simple scenario: a single defender and two locations, **simultaneous play**

Here, we can simply construct the payoff bimatrix for the game. We can then look for Nash Equilibria (NE). These are pairs of strategies (c, a) with expected payoffs (U, V) such that neither player can increase payoff by unilaterally choosing a different strategy; there is no c' that, when paired with a, yields a $U' > U$.

	Attack 1 a_1	Attack 2 a_2
c_1 Defend 1	$(U=0, V=0)$	$(U=U_2, V=V_2)$
c_2 Defend 2	$(U=U_1, V=V_1)$	$(U=0, V=0)$

Take a simple scenario: a single defender and two locations, **simultaneous play**

We can find a NE by looking for an equalizing strategy

Example – select a c such that V is independent of a:

Attack 1 gives $V_1 = c_1 \cdot 0 + (1 - c_1) \cdot V_1$
Attack 2 gives $V_2 = c_1 \cdot V_2 + (1 - c_1) \cdot 0$

Setting $V_1 = V_2$ gives $c_1 = \frac{V_1}{V_1 + V_2}, c_2 = \frac{V_2}{V_1 + V_2}$

Similarly, $a_1 = \frac{U_2}{U_1 + U_2}, a_2 = \frac{U_1}{U_1 + U_2}$

Hence, this particular (c, a) is a NE, as neither player can get a higher payoff by switching to something else unilaterally!

Let's try this out for a 3 location, 1 defender game

Visit the website https://crimemath.shinyapps.io/nash/

But, often in security games simultaneous play is not assumed.

- Instead, the defender is often assumed to be the "first player", committing to a strategy that is then known to the attacker(s) through observation/reconnaissance
- This is known as a Stackelberg Security Game (SSG)
- This problem is in some ways simpler than Nash Equilibrium – given c, a will be chosen to maximize $V(a;c)$, which then sets U. So, just maximize $U(c)$ to find the Stackelberg Equilibrium (SE)
- **Important**: for some c (such as equalizing strategies), there is no *unique* a that maximizes $V(a;c)$. In such a case, it is assumed that the attacker breaks ties optimally for the defender.

Take a simple scenario: a single defender and two locations, **defender plays first**

We can find a SE by first considering attacker best response

$V = a_1 (c_1 \cdot 0 + (1 - c_1) \cdot V_1) + a_2 (c_1 \cdot V_2 + (1 - c_1) \cdot 0)$

So, if $c_1 > \frac{V_1}{V_1 + V_2}$, choose $a_2 = 1$, giving $U = c_1 U_2 < -\frac{V_1}{V_1 + V_2} |U_2|$

if $c_1 < \frac{V_1}{V_1 + V_2}$, choose $a_1 = 1$, giving $U = (1 - c_1) U_1 < -\frac{V_2}{V_1 + V_2} |U_1|$

if $c_1 = \frac{V_1}{V_1 + V_2}$, choose a maximizing $U = -a_1 \frac{V_2}{V_1 + V_2} |U_1| - a_2 \frac{V_1}{V_1 + V_2} |U_2|$

Hence, choosing the c from our NE gives the highest U and is therefore the SE!

Let’s now try our 3 location, 1 defender game as a Stackelberg game

Visit the website
https://crimemath.shinyapps.io/nash/

There are some general results along these lines

1. Any SE utility is at least as good as any NE utility for the defender.
2. In security games (but not all games!), all NE are interchangeable. That is, if (c,a) and (c',a') are NE, then so are (c,a') and (c',a).
3. The attacker's V is **the same** for all (interchangeable) NE of a security game, but the defender's U is **not necessarily**.
4. Any SE's c is also a NE's c for that security game (but not necessarily vice versa), under the assumption that defenders can cover fewer targets than their resources would allow if they desire.
5. Hence, SE provide very reasonable solutions to security games, Stackelberg or not.

Korzhyk, Dmytro, et al. Journal of Artificial Intelligence Research (2011).

SSG are an active research area:

- In some applications, merely computing the SE is very challenging given the size of the domain.
- What if you are facing several different “types” of attackers, with different V_i?
- How do you actually know what the V_i are? Can you learn them by observing attacker behavior?
- What if some of our assumptions on the behaviors and capabilities of attackers are false?

Some (random?) references on these topics:

- Kiekintveld, et al. The 10th International Conference on Autonomous Agents and Multiagent Systems-Volume 3, 2011.
- Conitzer and Sandholm. Proceedings of the 7th ACM conference on Electronic commerce. ACM, 2006.
- Jain, et al. AAAI, 2010.
- Blum, et al. "Learning to Play Stackelberg Security Games.“, 2015.
- Balcan, et al. Proceedings of the Sixteenth ACM Conference on Economics and Computation. ACM, 2015.

Let's focus on the last questions, regarding the assumed attacker behaviors and capabilities.

We assumed attackers:

1. Are highly strategic
2. Plan attacks in advance
3. Are incapable/unwilling to make on-the-fly adjustments
4. Plan one "large" attack at a time
5. Play each "game" in a one-off manner

But, common criminals don't fit this mold. They:

1. Are slightly to moderately strategic
2. Perform attacks when opportunities arise
3. Make real-time decisions regarding attacks
4. May perform several "small" attacks in quick succession; multi-round games

We have therefore proposed the Opportunistic Security Game (OSG) to deal with them

- The game is multi-round for each attacker, with each round after the first occurring with probability α
- The attacker may be adjusting his strategy each round in response to his observations of the defender, causing him to move location to location
- The defender should therefore move around as well, so that her strategy is now an ergodic transition matrix T, for which the stationary coverage is the usual c
- Defender's goal: find T that maximizes U

We propose the following behavior for the attacker, who is assumed to know T and c

1. Attacker currently at location i makes observation o (sees defender or not) then constructs an estimated distribution $c_{est}(o)$ of the defender's *current* location using only c

   $$c_{est,i}(1)=1, c_{est,j}(1)=0 \;\forall\; j\neq i$$
   $$c_{est,i}(0)=0, c_{est,j}(0)=c_j/(1-c_i) \;\forall\; j\neq i$$

2. For any potential next attacker location j at an integer temporal distance denoted $|j-i|$, the attacker can estimate the probability the defender will be there when the attacker would arrive: $(T^{|j-i|})_j\, c_{est}(o)$
3. The expected value of j from i given o is then

   $$V_j(i,o)=V_j\left[1-(T^{|j-i|})_j\, c_{est}(o)\right]$$

We propose the following behavior for the attacker

4. Given the $V_j(i,o)$, the attacker chooses strategy

$$a_j = V_j(i,o)^{\lambda} / \sum_k V_k(i,o)^{\lambda}$$

Rationality parameter $\lambda \geq 0$

This is an example of bounded rationality, a sort of fudge factor to account for the fact that attackers may not always be able to make the best decision. For $\lambda=0$ choices are random, and as $\lambda \rightarrow \infty$ we get closer to the optimal choice.

5. The attacker then chooses a specific j according to a, and the process repeats with probability α.

We express the final (Markov process) problem thusly:

- Let state space be the set of all pairs of locations $s=(l_d, l_a)$, where l_d is the defender's location and l_a is attacker's location. s determines o.
- Let M be the transition matrix in state space:

$$M_s^{s'} = T_{l_d}^{l_d'} \, a_{l_a}^{l_a'}(l_a, o(s))$$

- Let R be the vector of utilities for the defender expressed in state space; $R_s = U_{l_a}$ for $o(s)=0$ and $R_s = 0$ for $o(s)=1$

We express the final (Markov process) problem thusly:

- Then, starting from initial state space vector x_0, which may depend on c, the defender's expected utility is

$$U = R \cdot (I + \alpha M + \alpha^2 M^2 + \cdots) x_0 = R \cdot (I - \alpha M)^{-1} x_0$$

- Goal: choose T to maximize U.
- Note that this is a nasty problem!
  - Nonlinear constrained optimization
  - M is order N^2 for one defender, N^{1+D} for D defenders

Let's take a pause and play a 3 location, 1 defender OSG, assuming you all are very rational

Visit the website
https://crimemath.shinyapps.io/osgame/

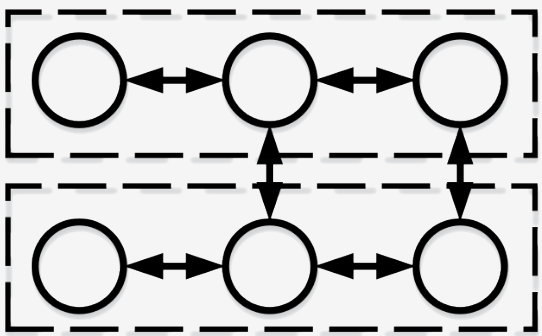
Let’s try that again, assuming you all are not so rational…

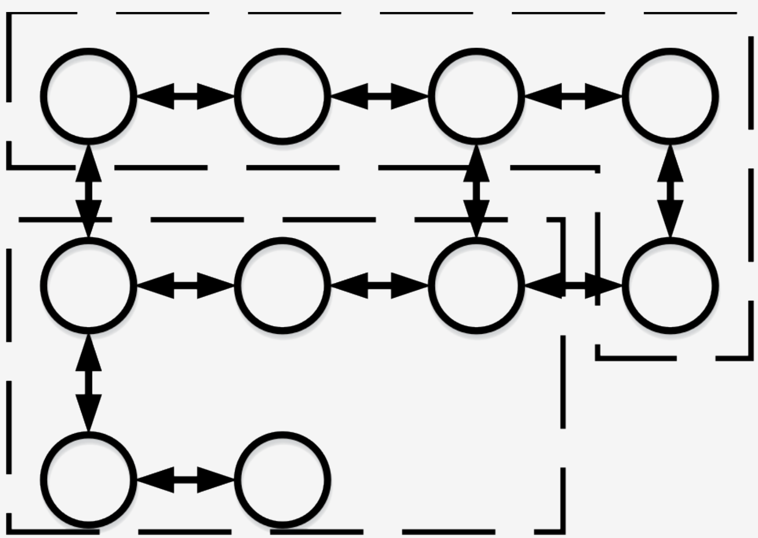
Visit the website
https://crimemath.shinyapps.io/osgame2/

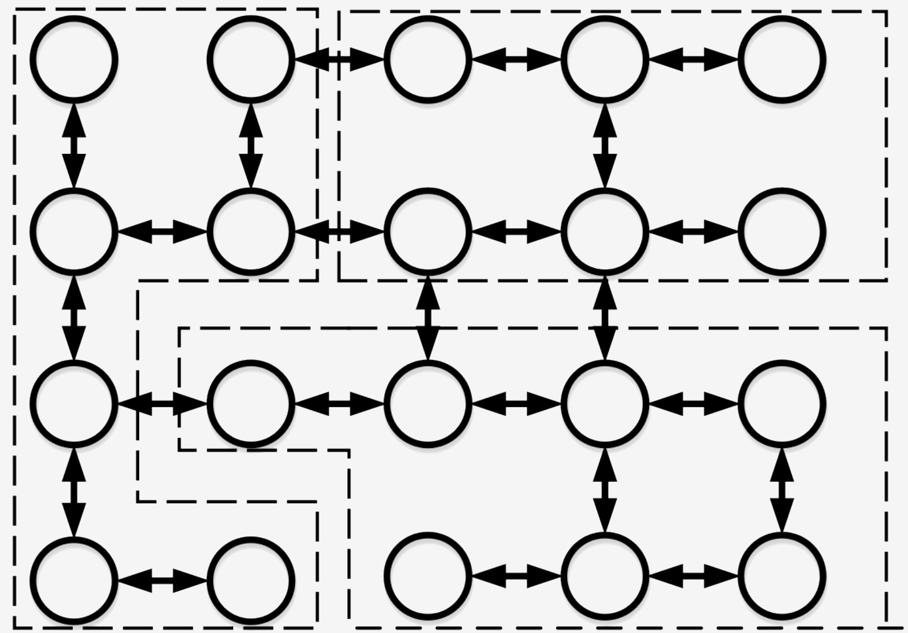
Solving an OSG is computationally challenging. But, an approximate problem is much simpler…

- Consider new state space with members $s_c = (o, l_a)$
- This space has size of order N, *regardless of number of defenders* D
- Here, instead of including l_d explicitly in the state space, we simply estimate it given l_a and o, the same way a criminal would
- The rest of the simplified problem is analogous to the full problem previously presented

Some results for artificial settings, comparing the full and approximate algorithm

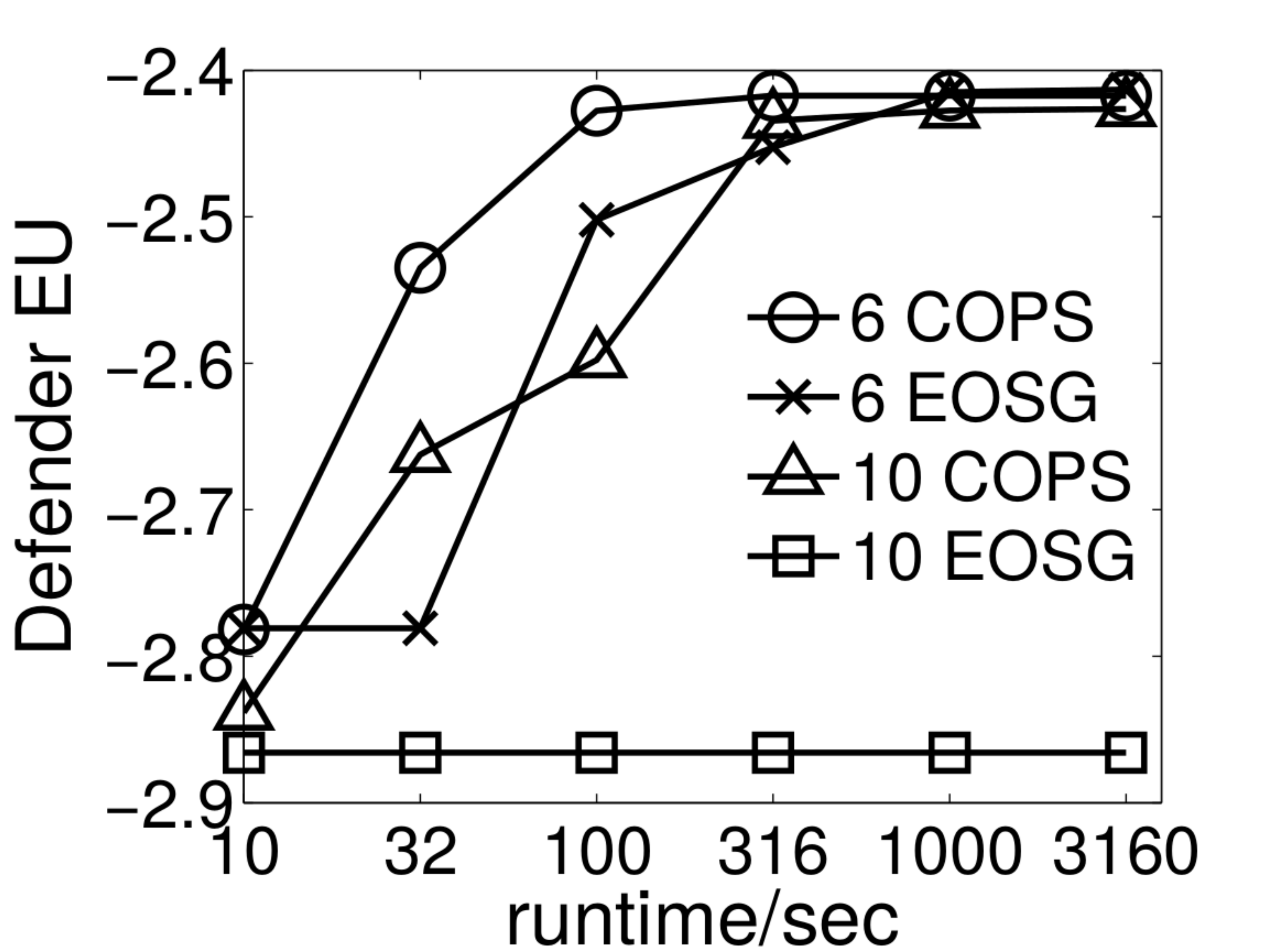




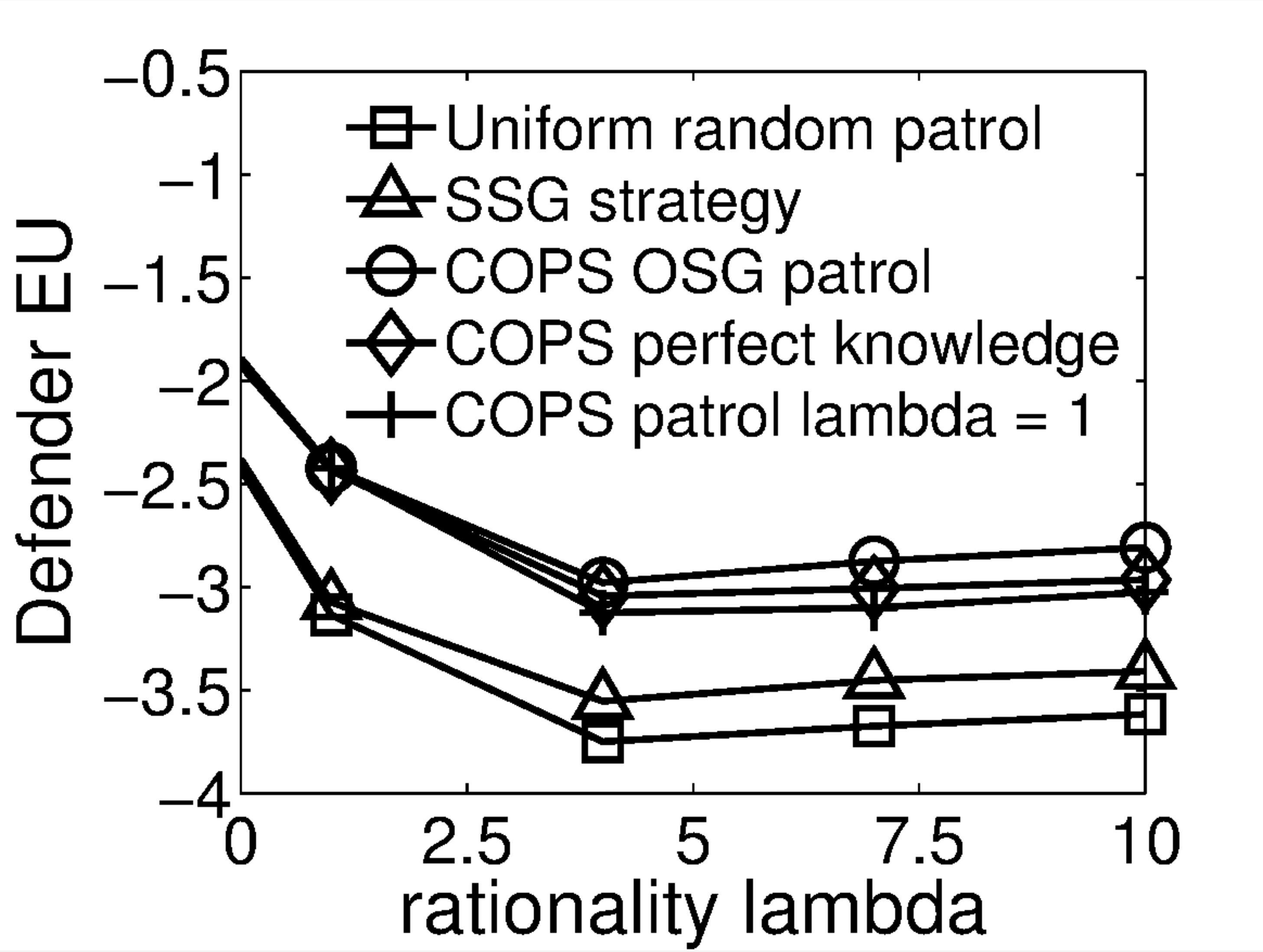


Each run performed for 30 different random V, each $V_i \in U(0,1]$ All $U_i = -1$

Defender Utility versus runtime

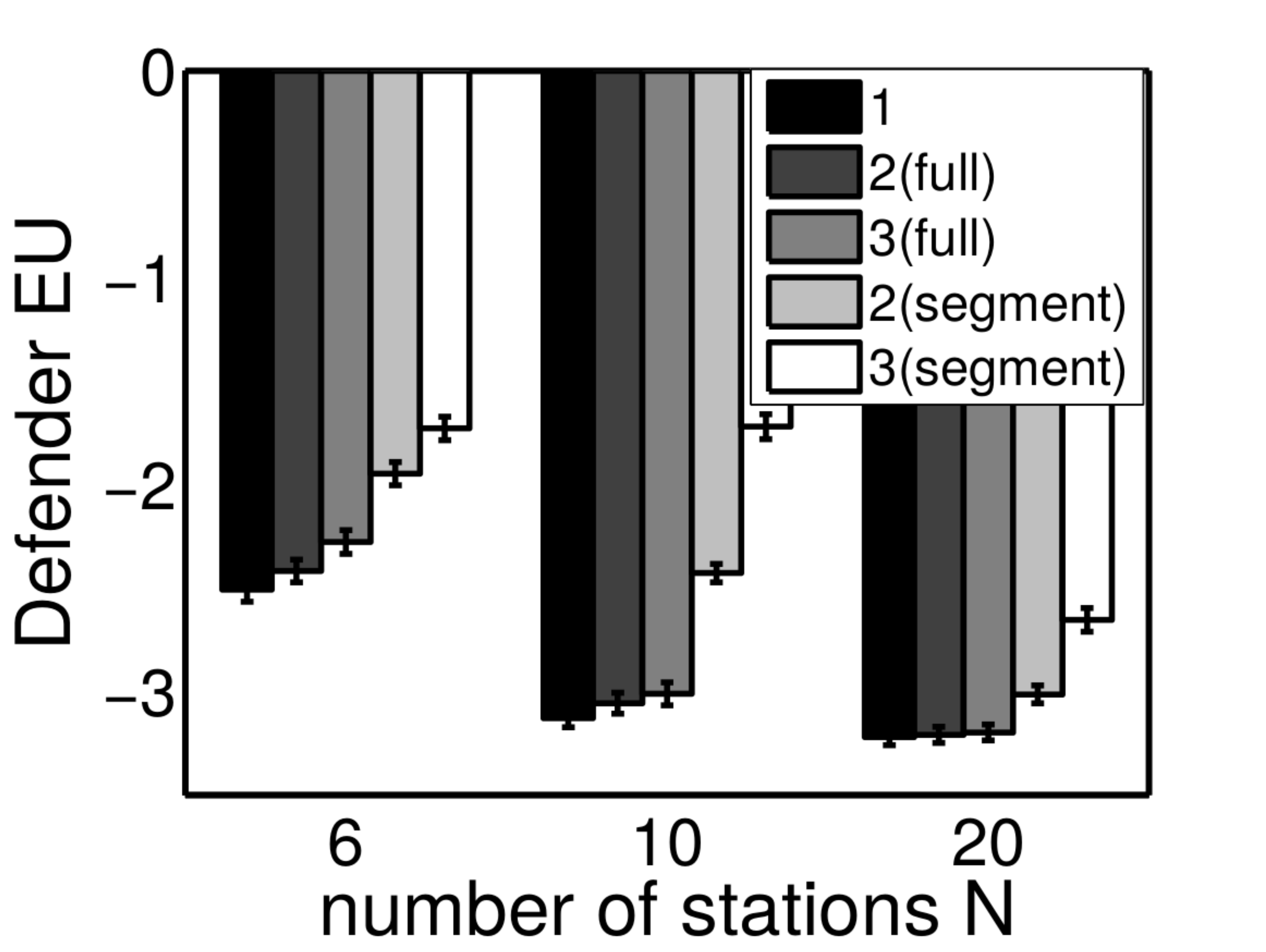


Defender Utility versus rationality level λ



6 locations and 2 defenders

Defender utility for differing D



Security games aren’t just theoretical, and have been deployed in many domains.

ARMOR - For the LAX Airport Police

PAWS - For Wildlife Conservation

OSG - For Opportunistic Crime

PROTECT - For the US Coast Guard

ARMOR-FISH - For the US Coast Guard

IRIS - For the Federal Air Marshal's Service

GUARDS - For the Transportation Security Administration

TRUSTS - For the Los Angeles Sheriff's Department

See http://teamcore.usc.edu/projects/security/ for more details

So, where does this leave us?

- As we've (hopefully) illustrated, the mathematics used to predict and prevent crime comes from many different subfields, each with interesting mathematical problems to explore
- Consequently, there are probably many new methods that will be developed in this field as it progresses
- There is a strong track record of field implementation in this area, so impact can be large

Acknowledgements

- Andrea Bertozzi (UCLA Math), Jeff Brantingham (UCLA Anthropology), George Tita (UC Irvine Criminology), Milind Tambe (USC Computer Science)
- Alexey Stomakhin, Joe Woodworth, Eric Lewis, Chao Zhang (former grad students)
- Many REU and RIPS summer students
- LAPD, Kent PD, Sean Malinowski, PredPol Inc.
- NSF, ARO, AFOSR, ONR for funding

THANK YOU