AI*IA Awards: In Memory of Leo, Lesmo Award for Best Thesis

Surveilling and protecting valuable targets exploiting a spatially uncertain alarm system

Giuseppe De Nittis
Supervisor: Nicola Gatti
Co-Supervisor: Nicola Basilico

14th Conference of the Italian Association for Artificial Intelligence
25th September 2015, Ferrara

Anavilhanas natural reserve (about 4000 Km^2)

Flying drone

A spatially uncertain signal

One signal, multiple targets

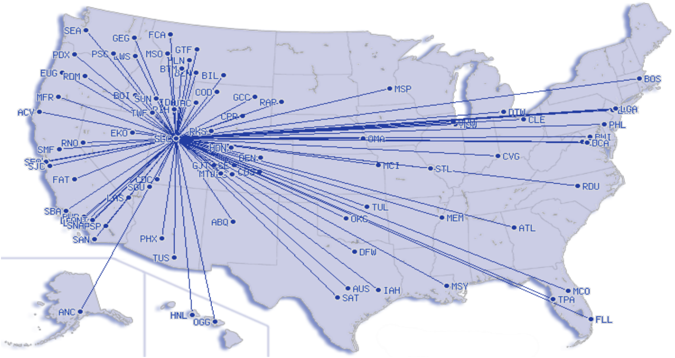
Security games

The Defender controls resources to protect the environment

The Attacker tries to compromise some areas without being detected

History

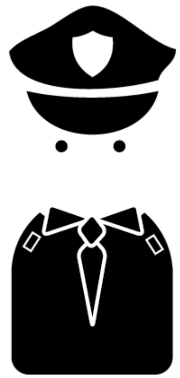
Los Angeles, 2008
AAAI, AAMAS

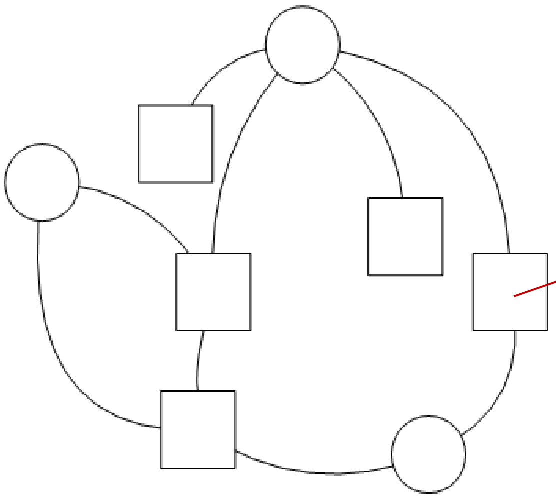


U.S. domestic flights, 2009
AAMAS

Milan, 2015

The model



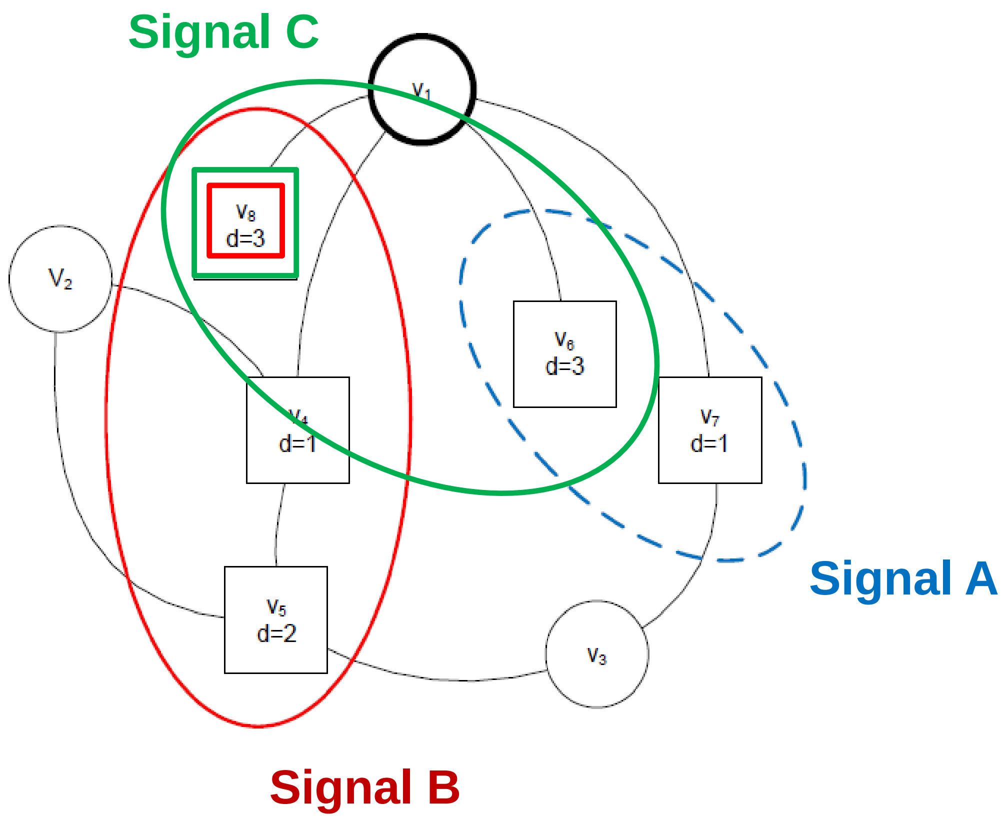


Target t:

- $\pi(t)$: value
- d(t): penetration time

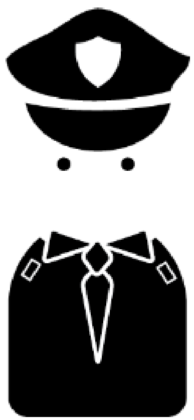
The alarm system

When a target is attacked, a spatially uncertain signal is generated



The actions

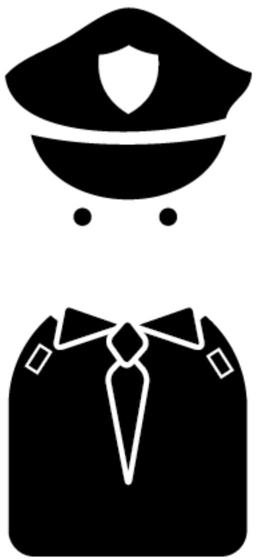
At any stage of the game:



The Defender decides where to go next

The Attacker decides whether to attack a target or to wait

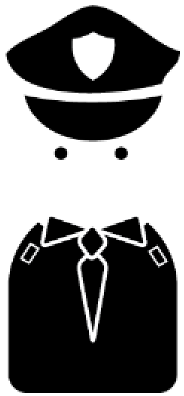
Utilities



$U(\{c, t\}) = (1, 0)$

$U(\{c, t\}) = (1 - \pi(t), \pi(t))$

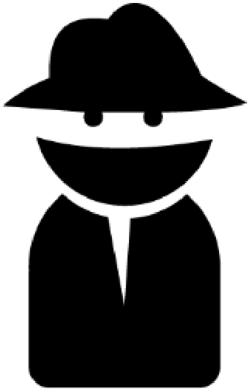
Solving the game



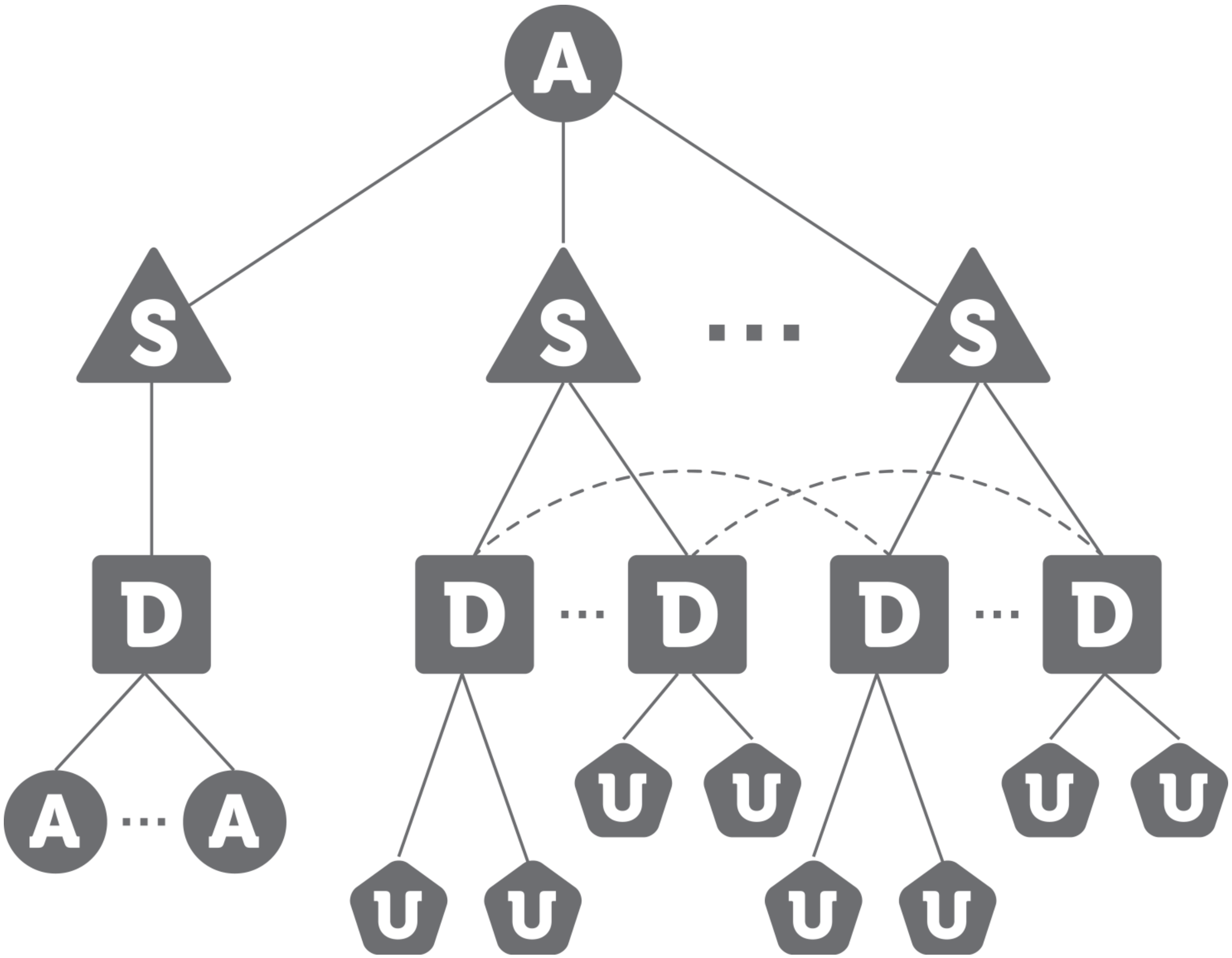
The Attacker can observe the Defender's strategy and knows it

The Defender's strategy is common knowledge of the game

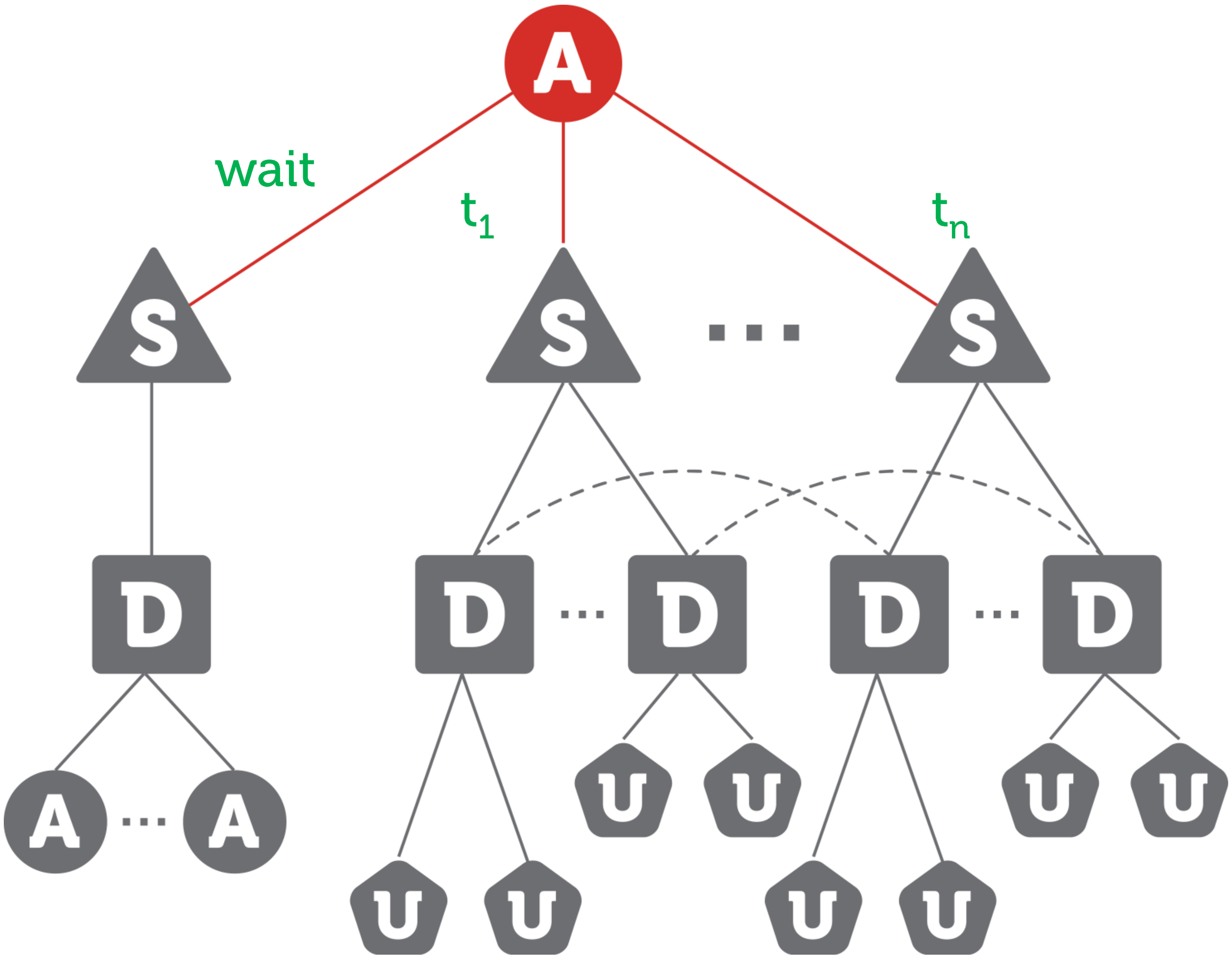
We adopt a Stackelberg paradigm, reducing to a maxmin equilibrium



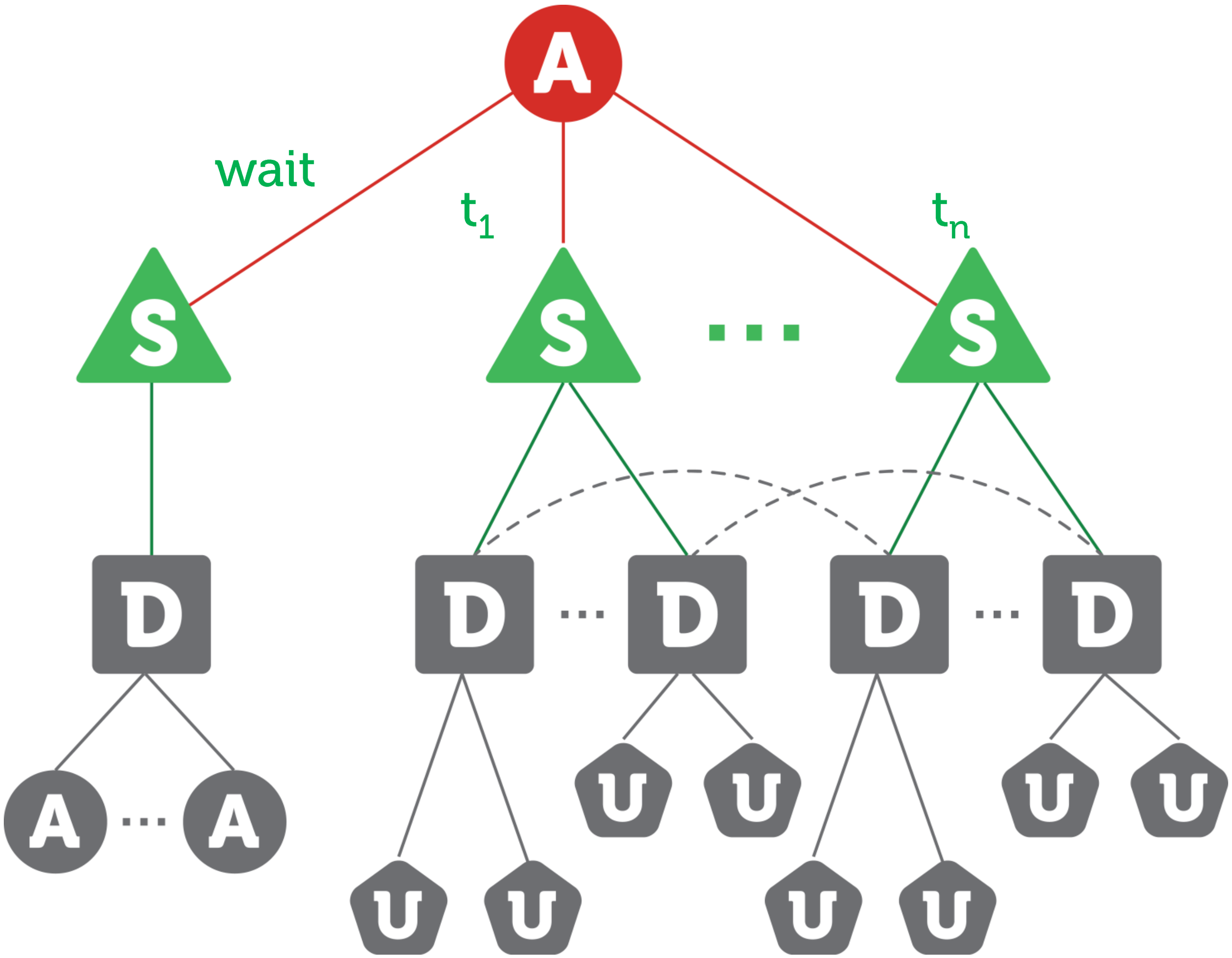
Interactions



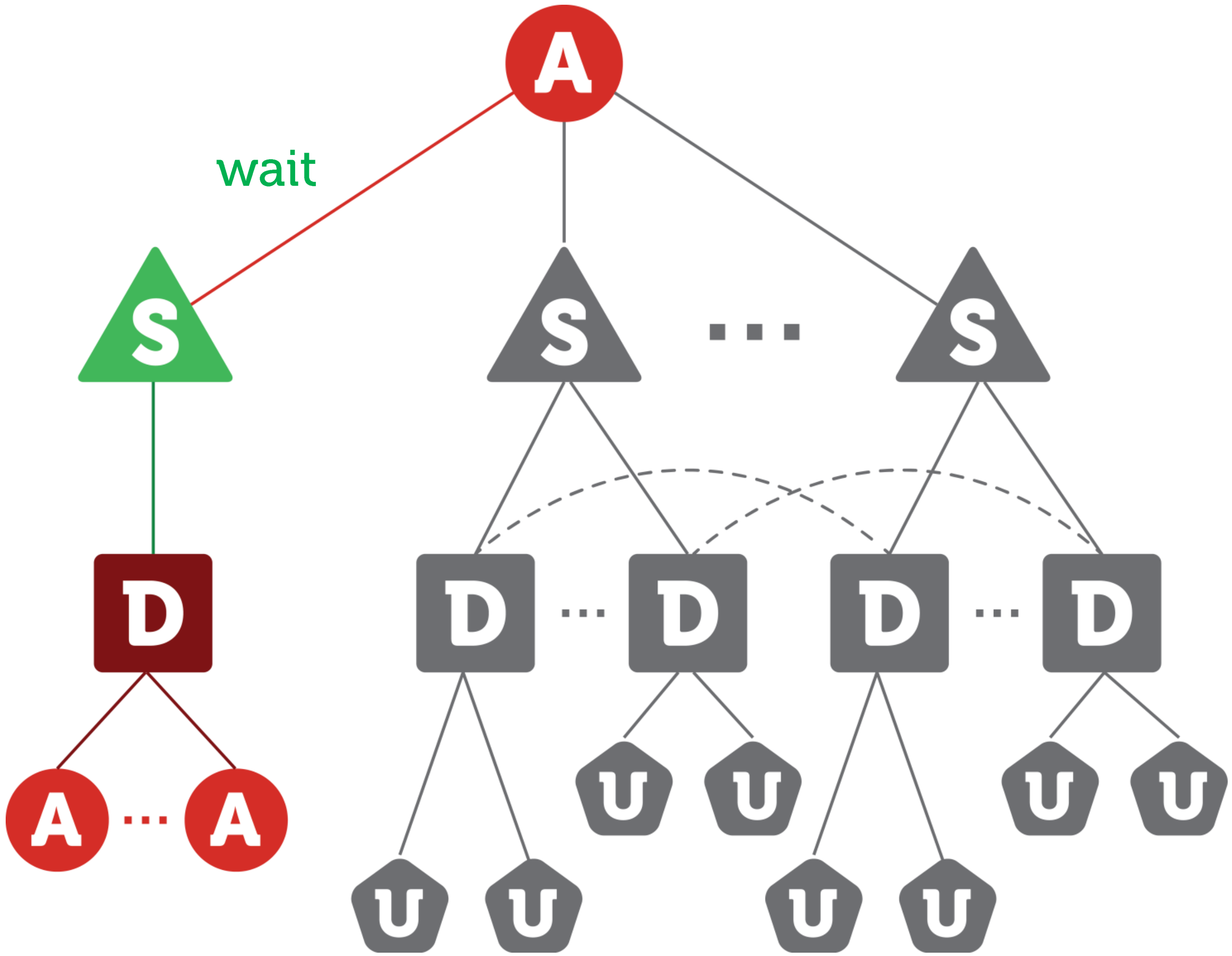
The attacker's action



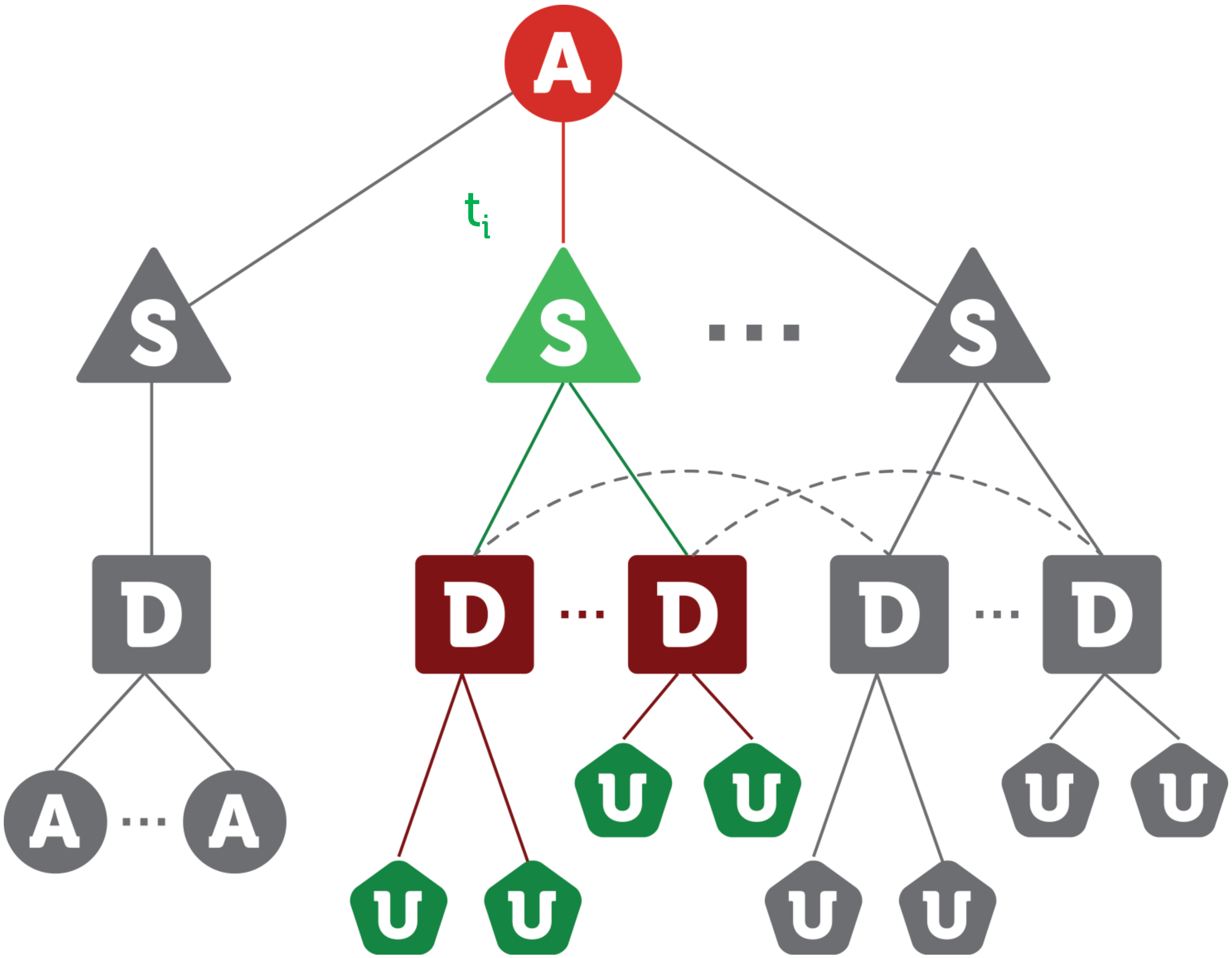
The alarm system



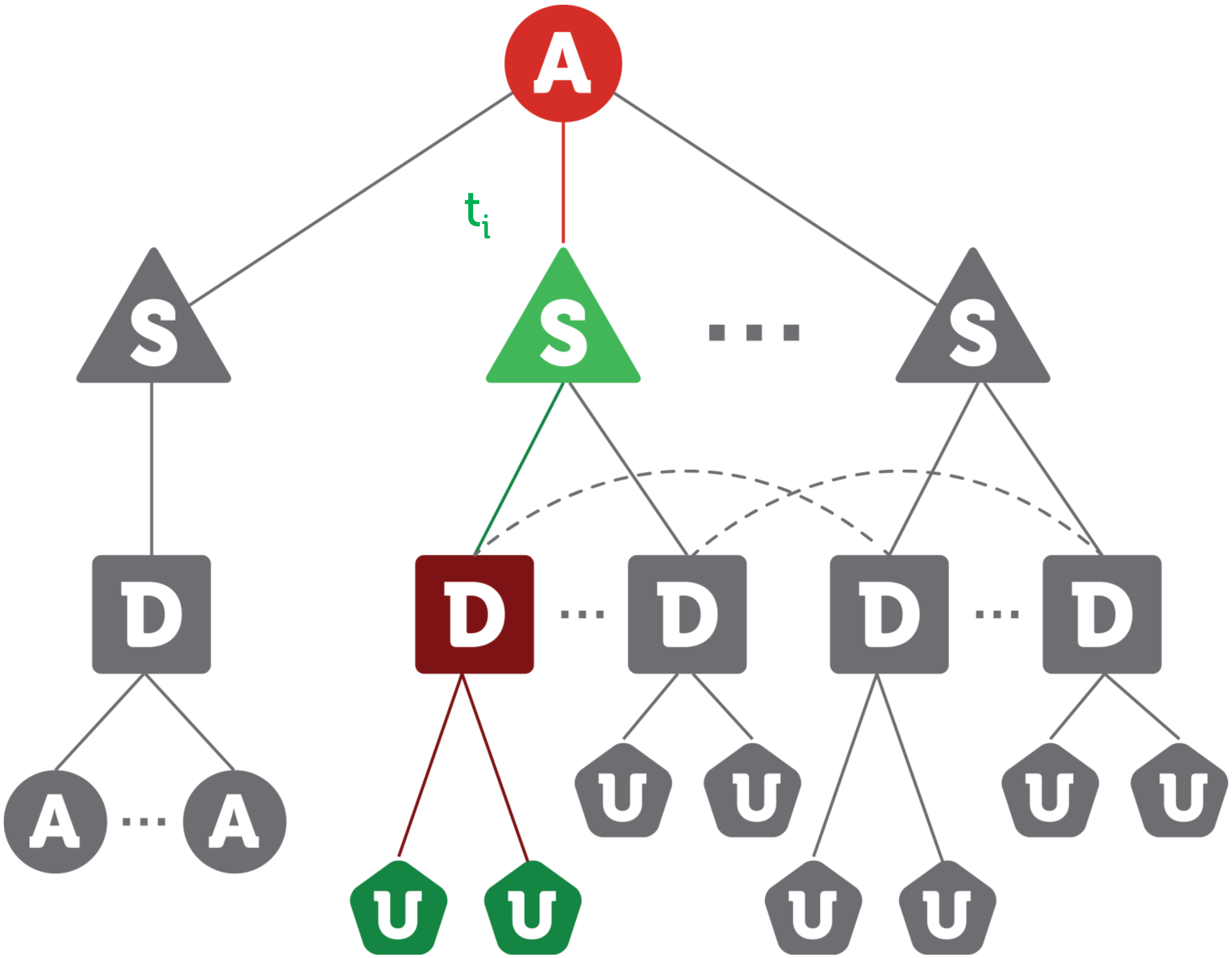
Patrolling Game (PG)



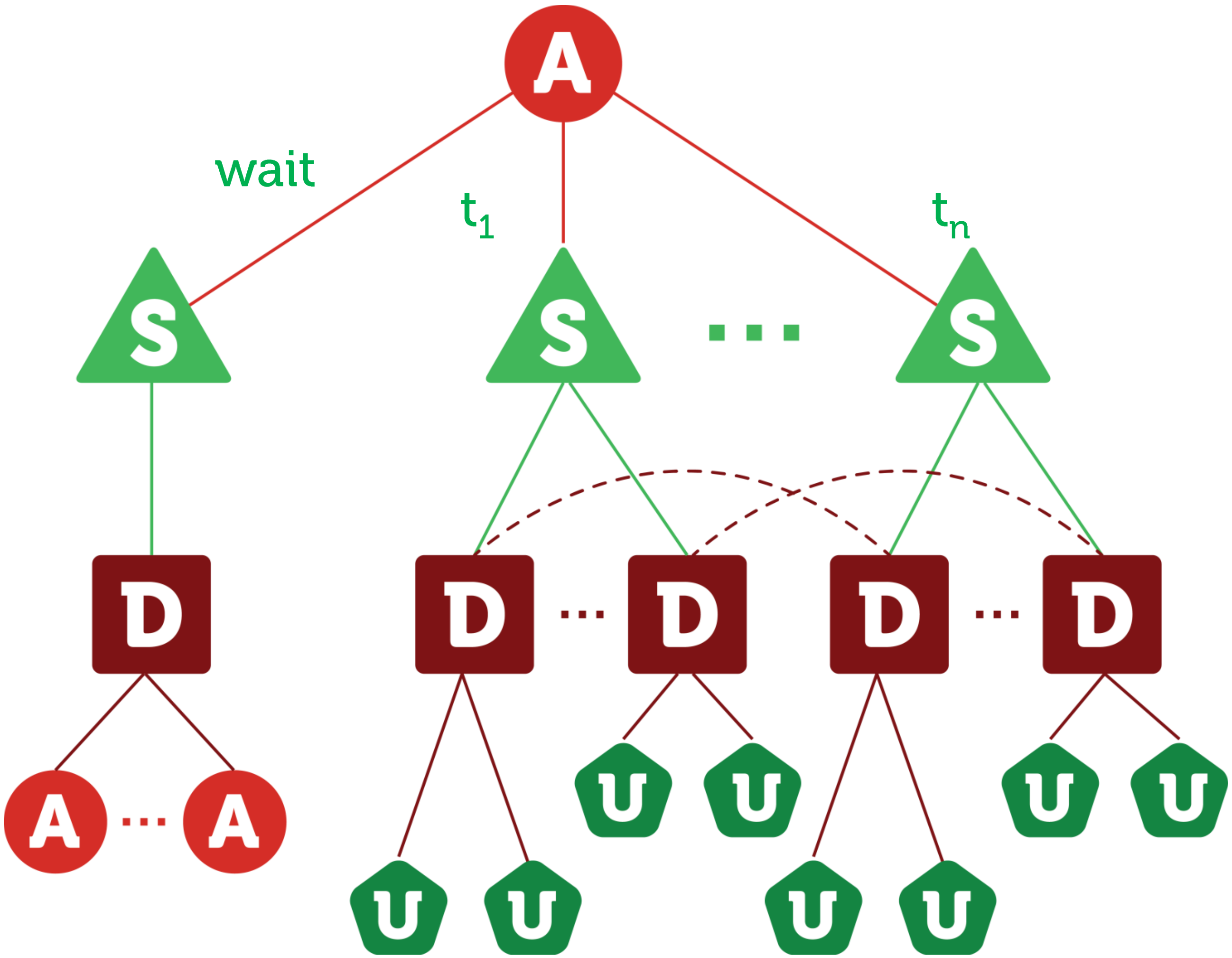
Signal Response Game (SRG)



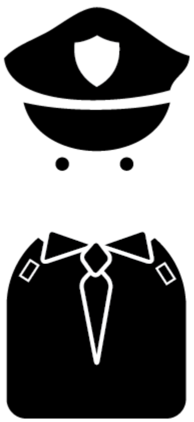
SRG-v



Interactions



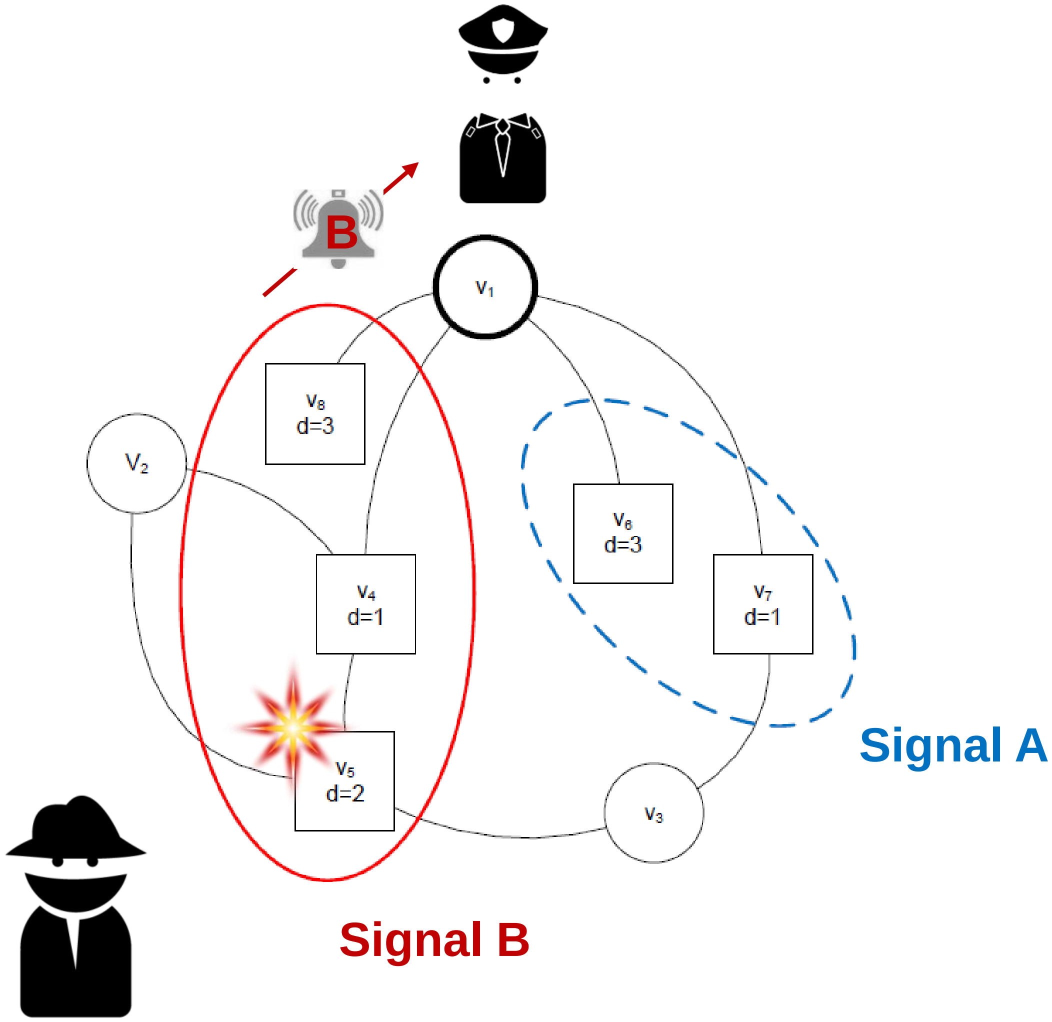
Two phases of the game



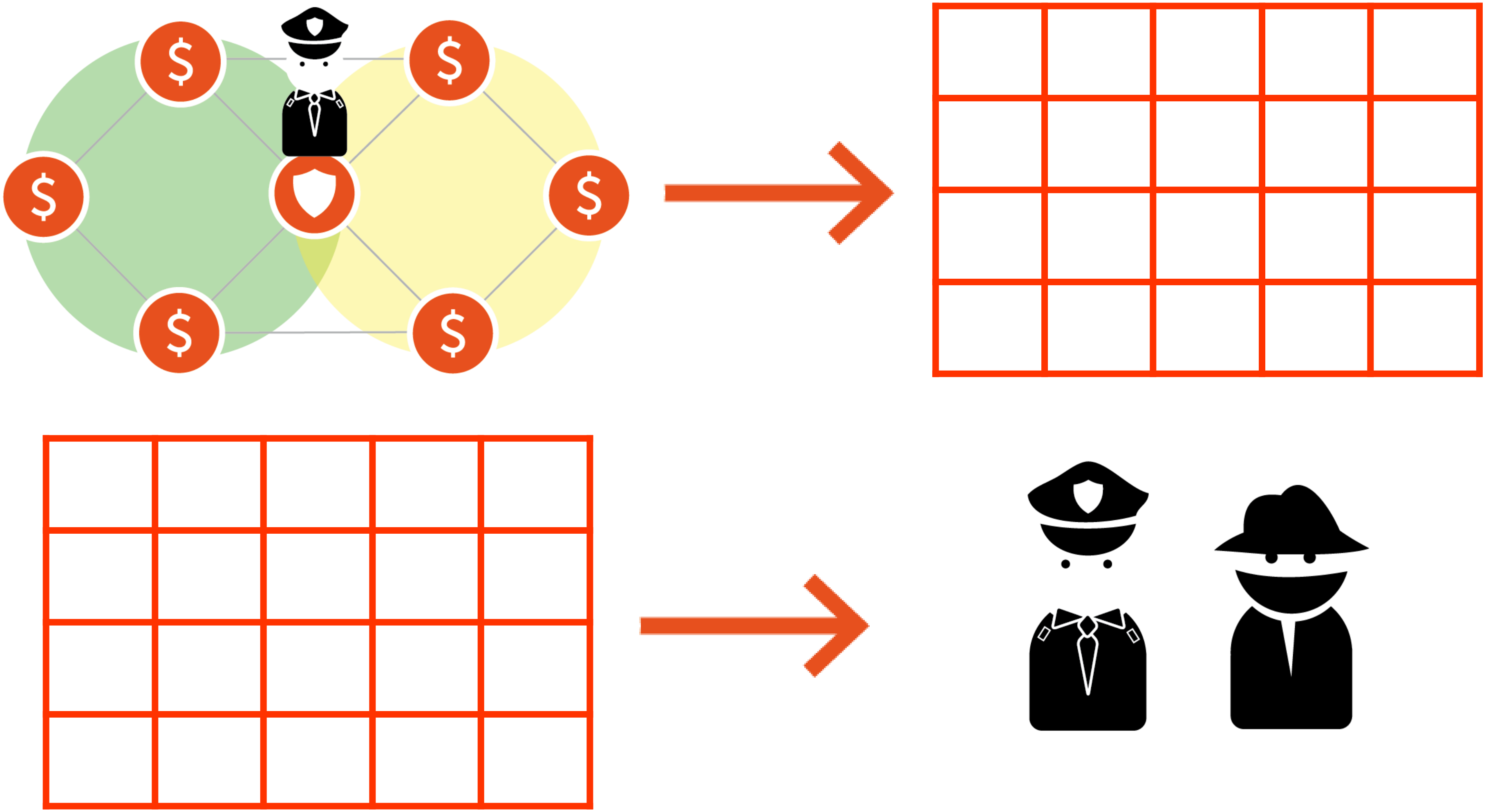
Normal patrolling

Signal response

The SRG-v



The two halves of the SRG-v problem

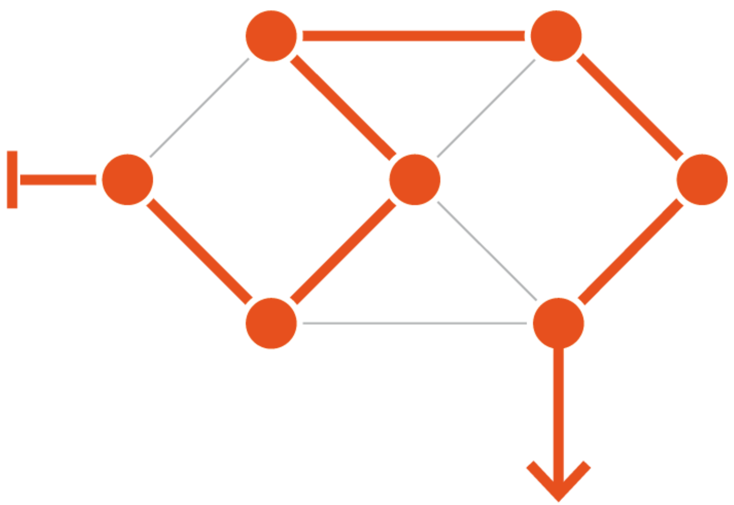


A hard task: SRG-v on arbitrary graphs

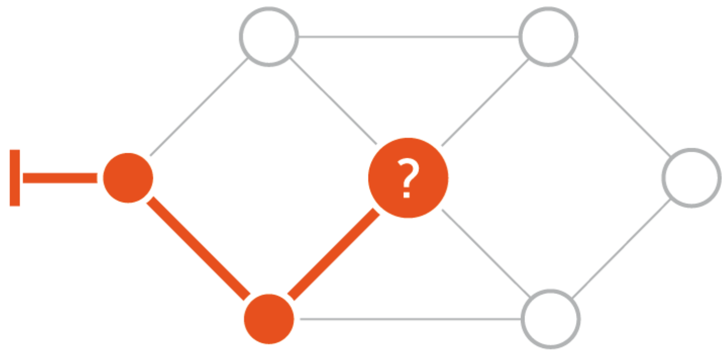
INSTANCE: an instance of SRG-v

QUESTION: is there any σ^D such that $g_v \leq k$?

k-SRG-v is strongly NP-hard even with |S| = 1.



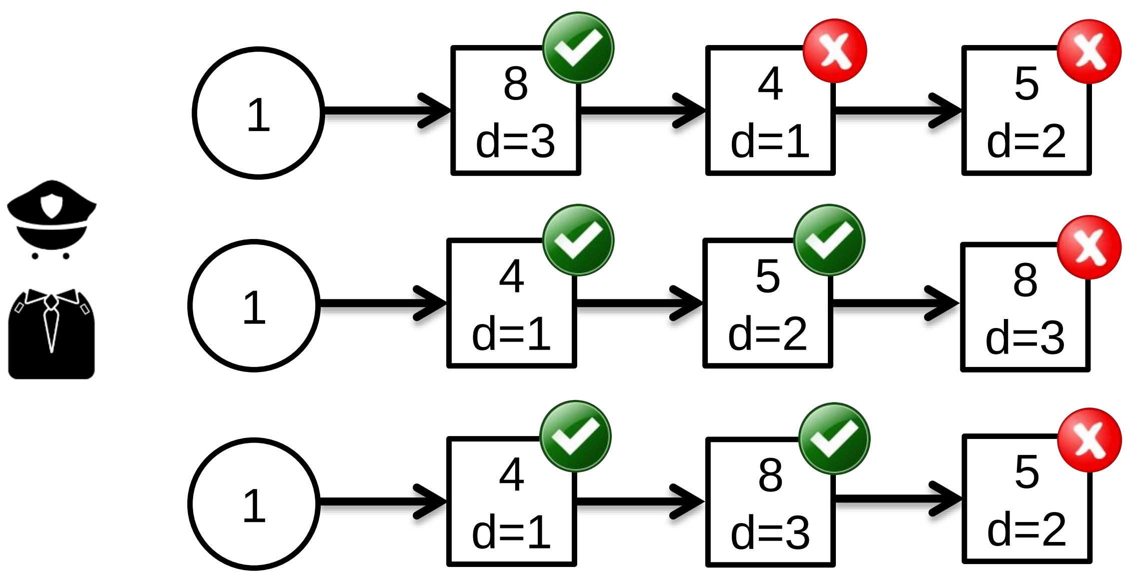
Hamiltonian Path

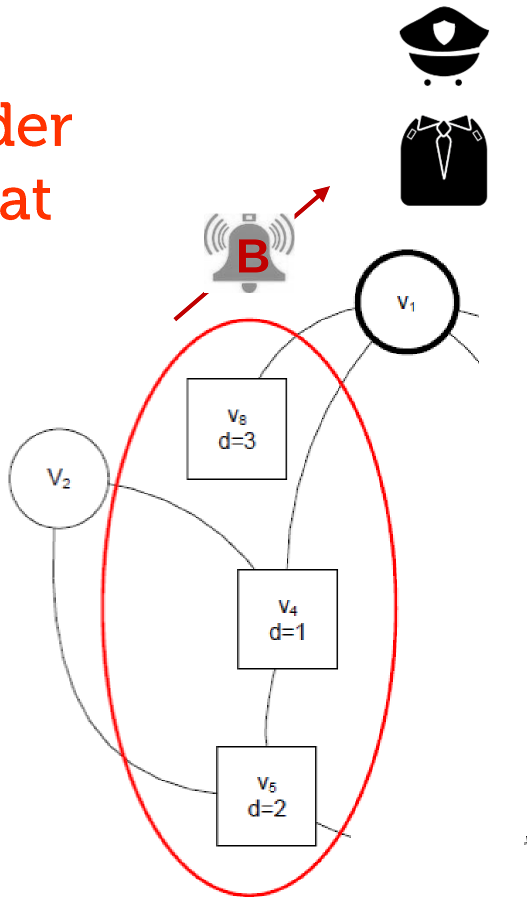


SRG-v

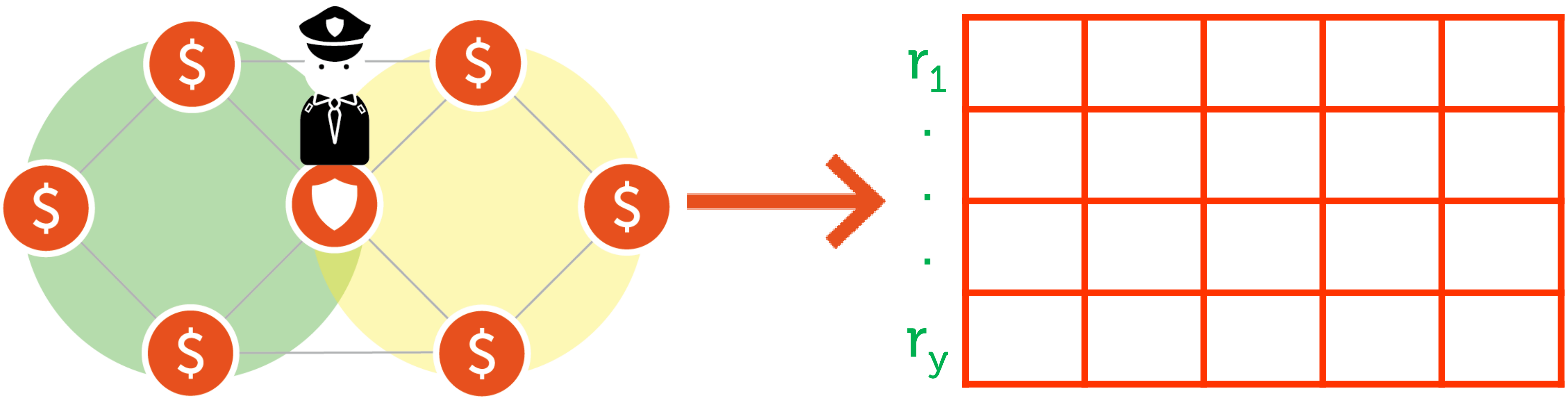
Covering route

A permutation of targets that specifies the order of first visits (covering shortest paths) such that each target is first-visited before its deadline





Building the game

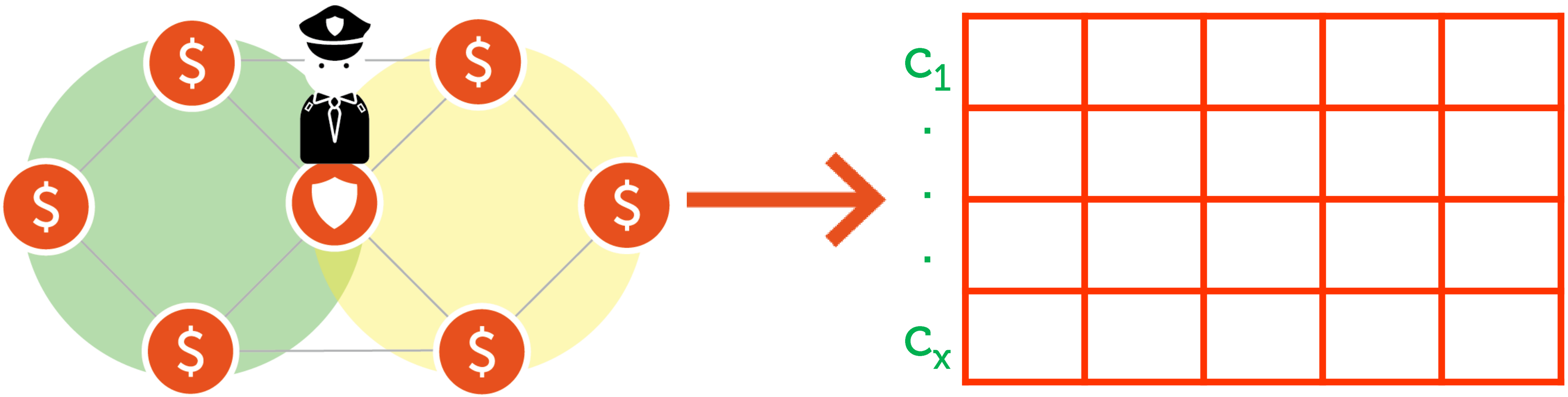


Complexity: $O(n^n)$

Covering sets

Can we consider covering ***sets***?

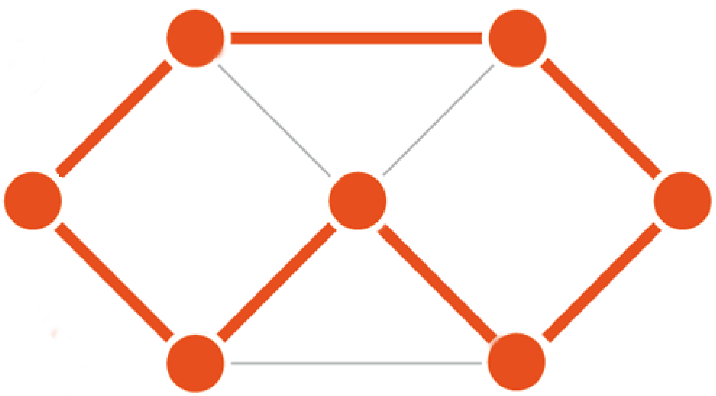
From $\langle t_1, t_2, t_3 \rangle$ to $\{t_1, t_2, t_3\}$

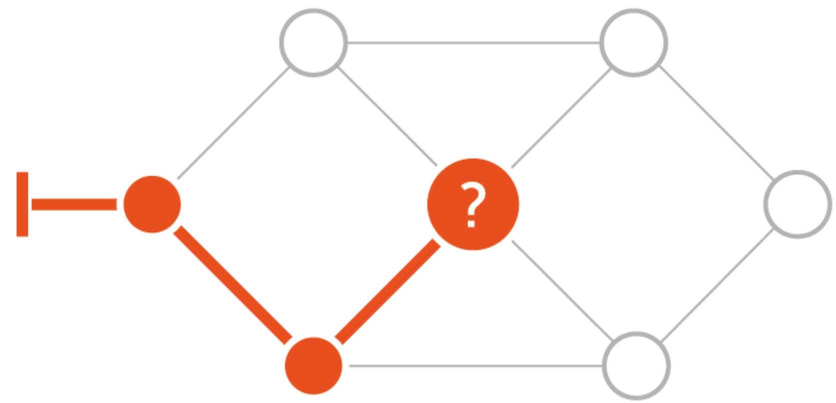


Complexity: $O(2^n)$

Approximating SRG-v game value

The optimization version of k-SRG-v is APX-hard even for very simple instances

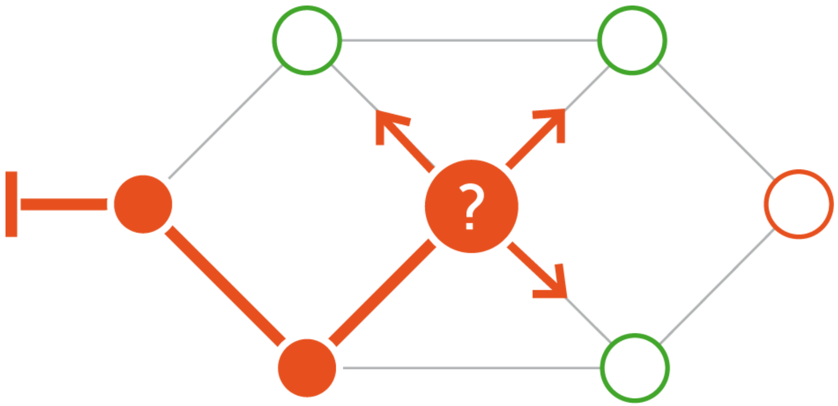

TSP(1,2)



SRG-v

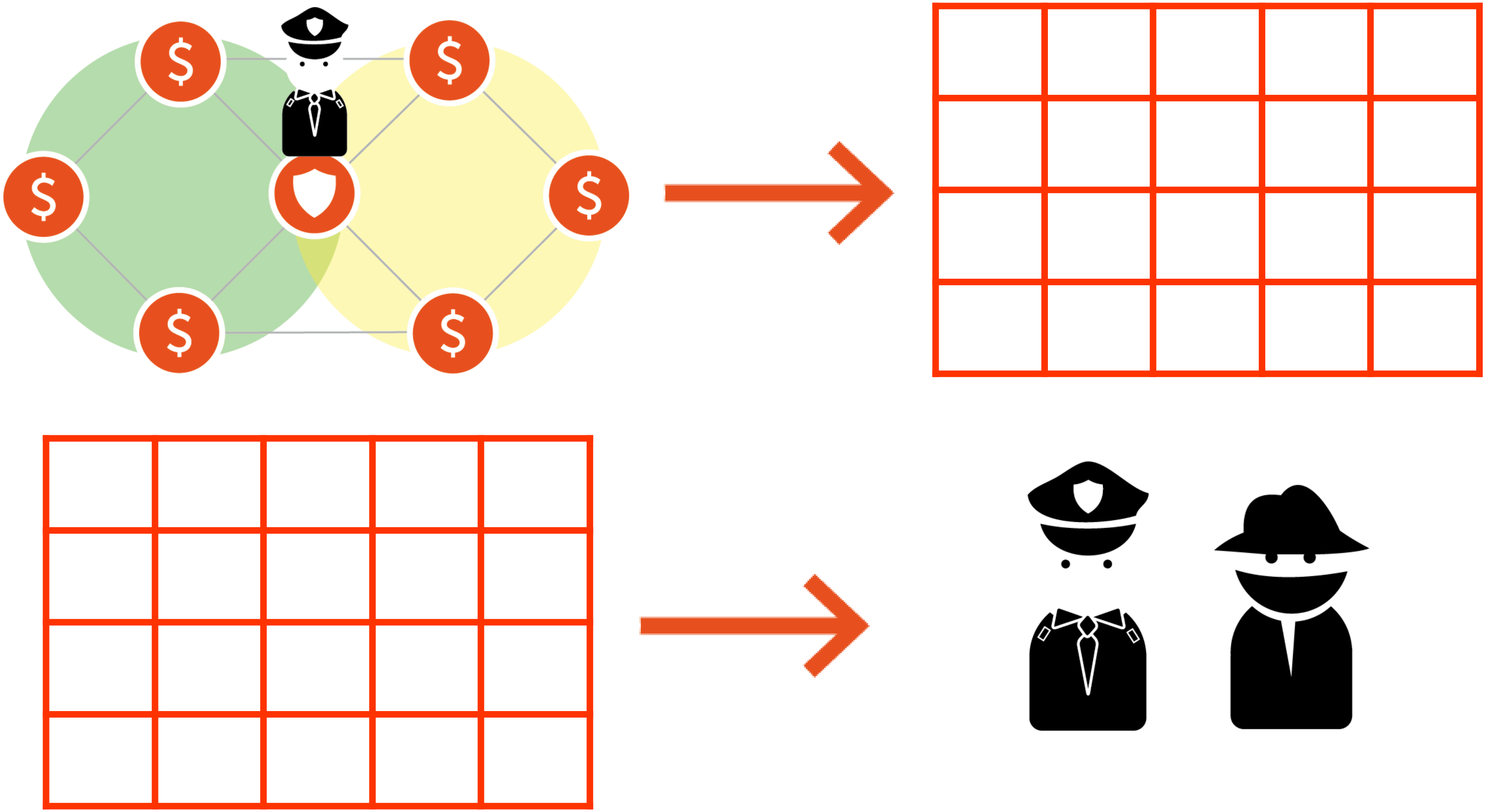
Our algorithm

We simultaneously build covering sets and the shortest associated covering route

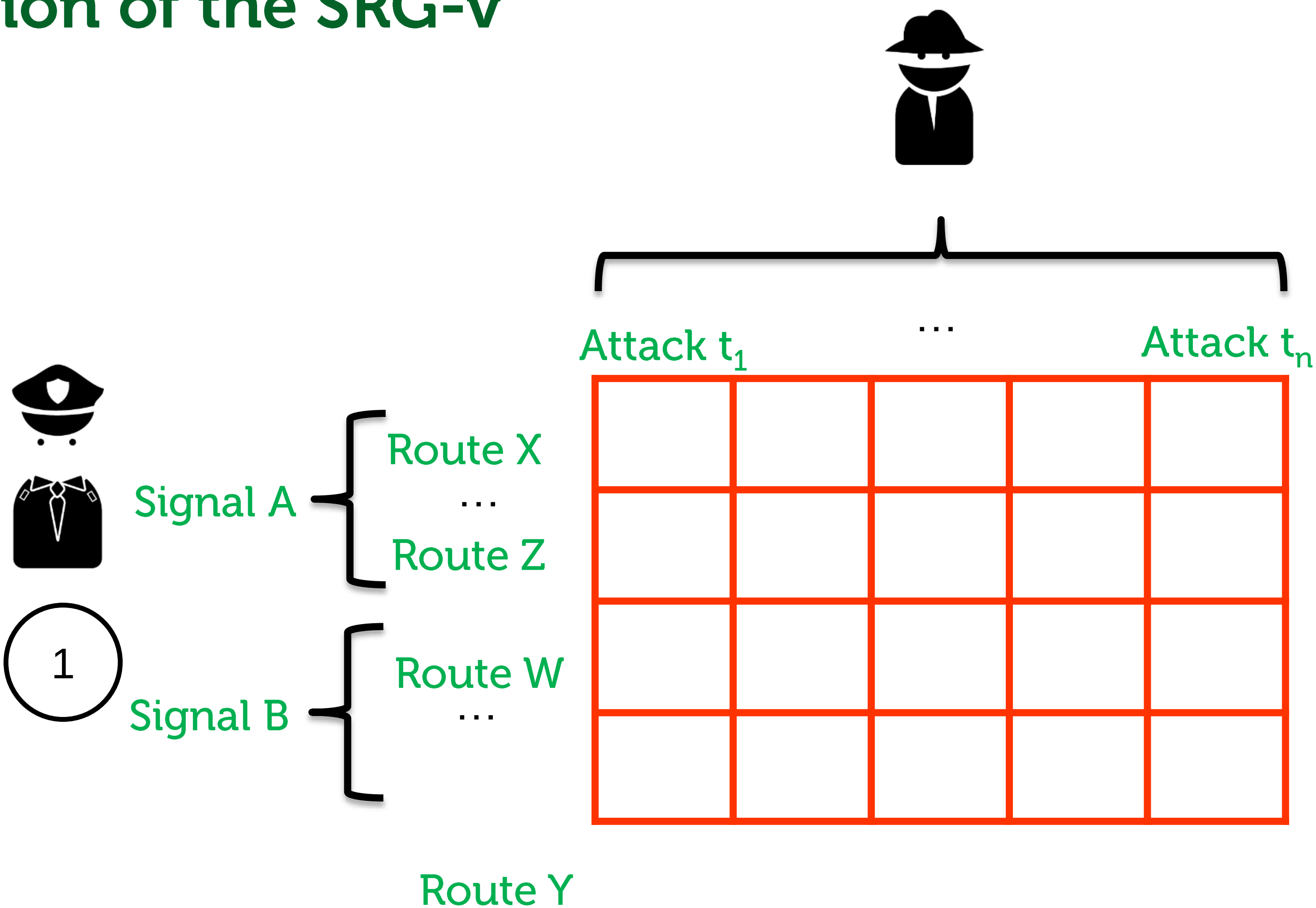


Dynamic programming inspired algorithm: we can compute all the covering routes in $O(2^n)$

A hard task



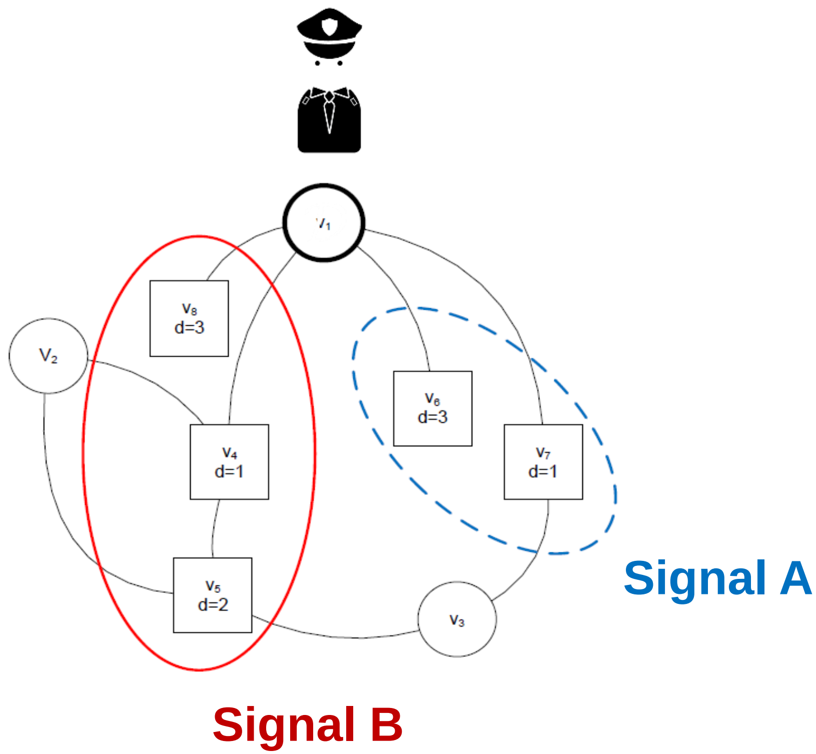
Solution of the SRG-v



Complexity: $O(n^c)$

The Patrolling Game

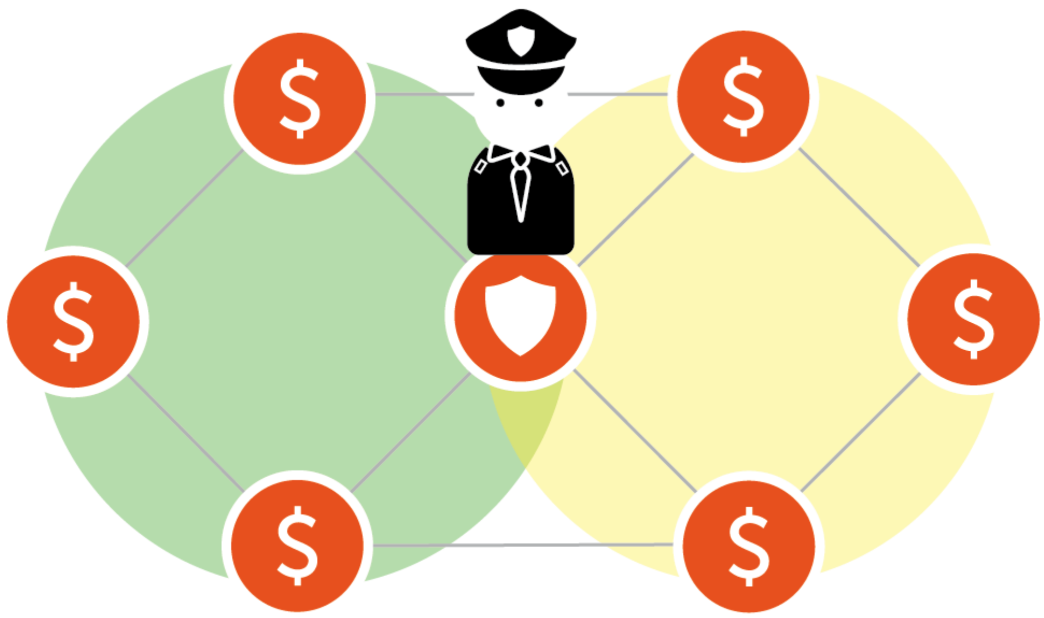
What to do when no signal is received?



The Attacker can observe the position of the Defender

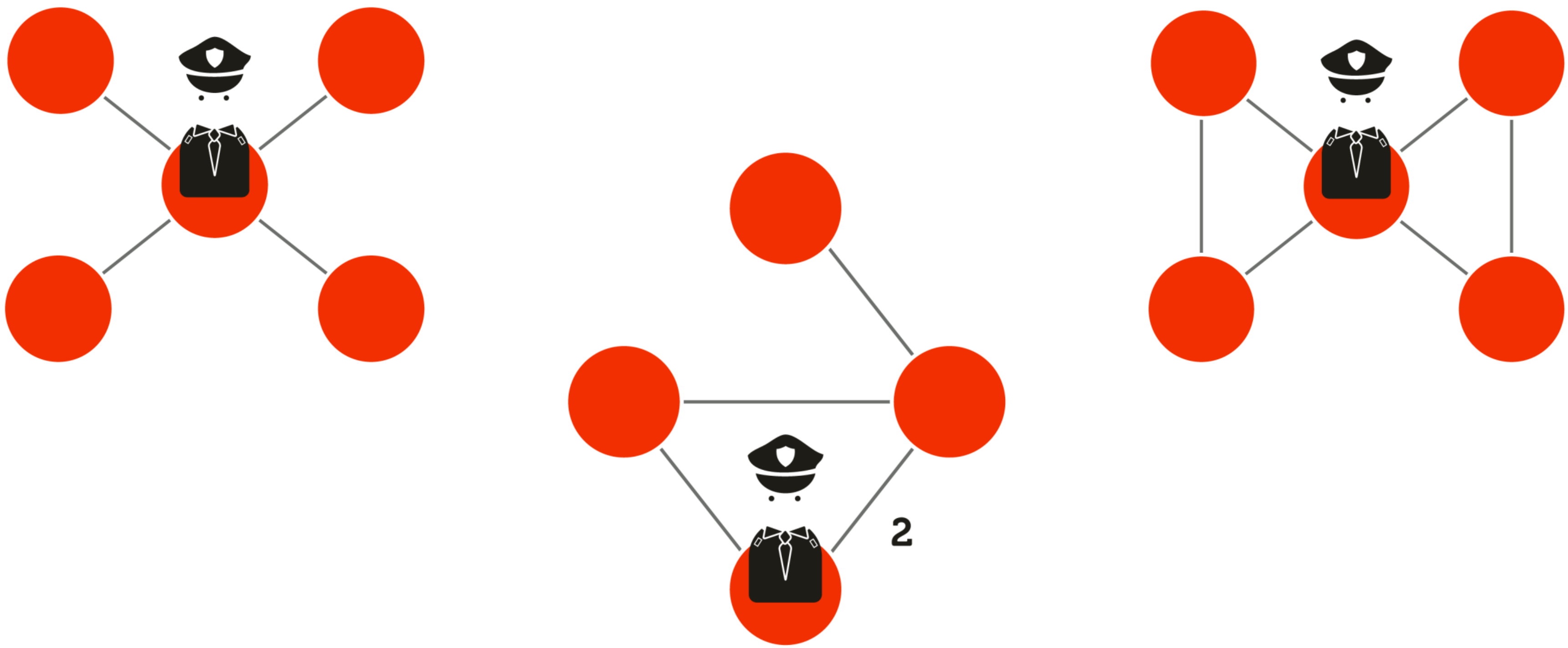
Stand still

Without false positives and missed detections, if the alarm system covers all the targets, then any patrolling strategy is dominated by the placement in v*



Special instances

There exist Patrolling Games where staying in a vertex, waiting for a signal, and responding to it is the optimal patrolling strategy for D even with a missed detection rate $\alpha = 0.5$



Experimental campaign

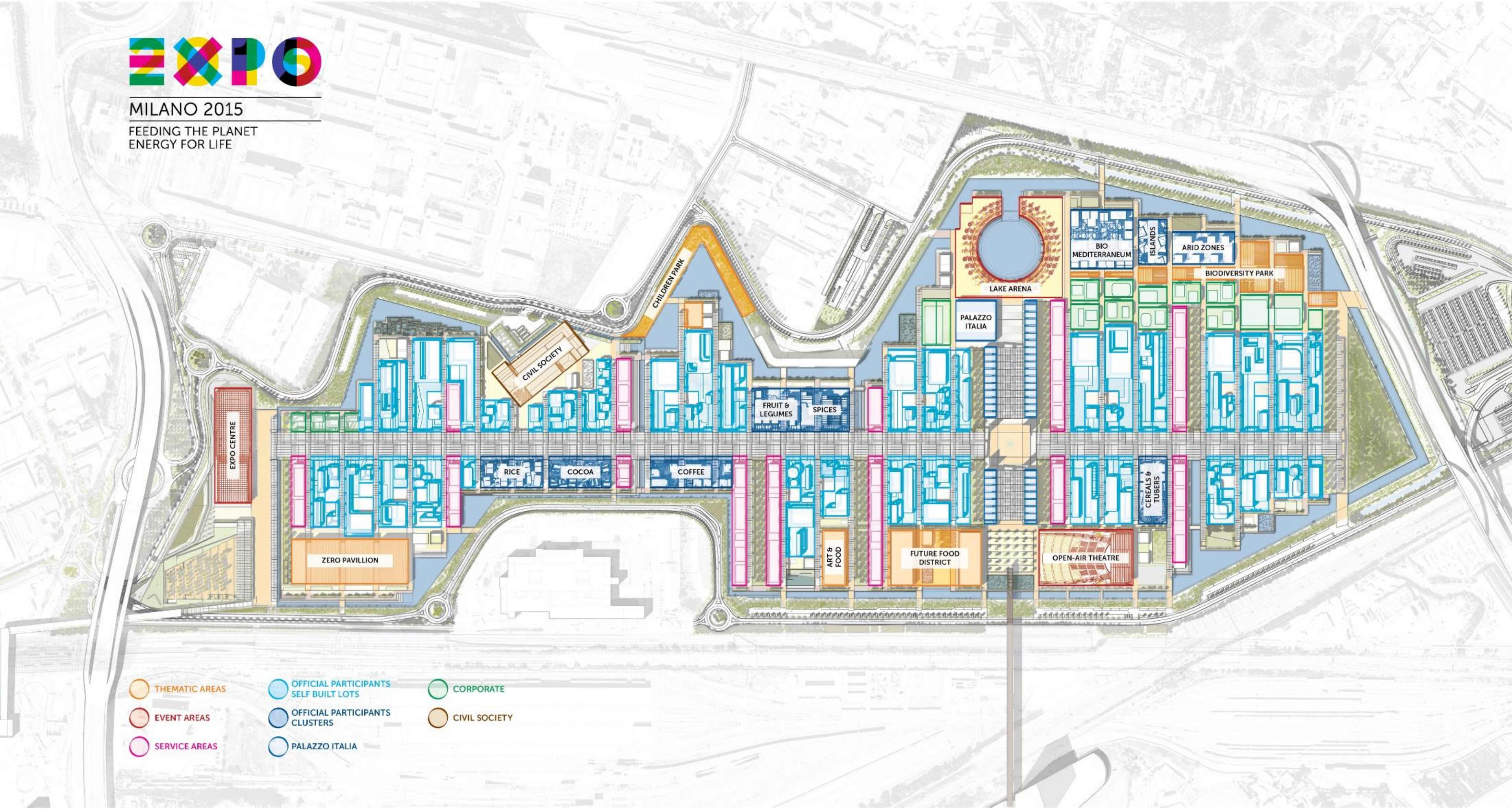
Hard instances: up to 20 targets

- Require the computation of an Hamiltonian Path

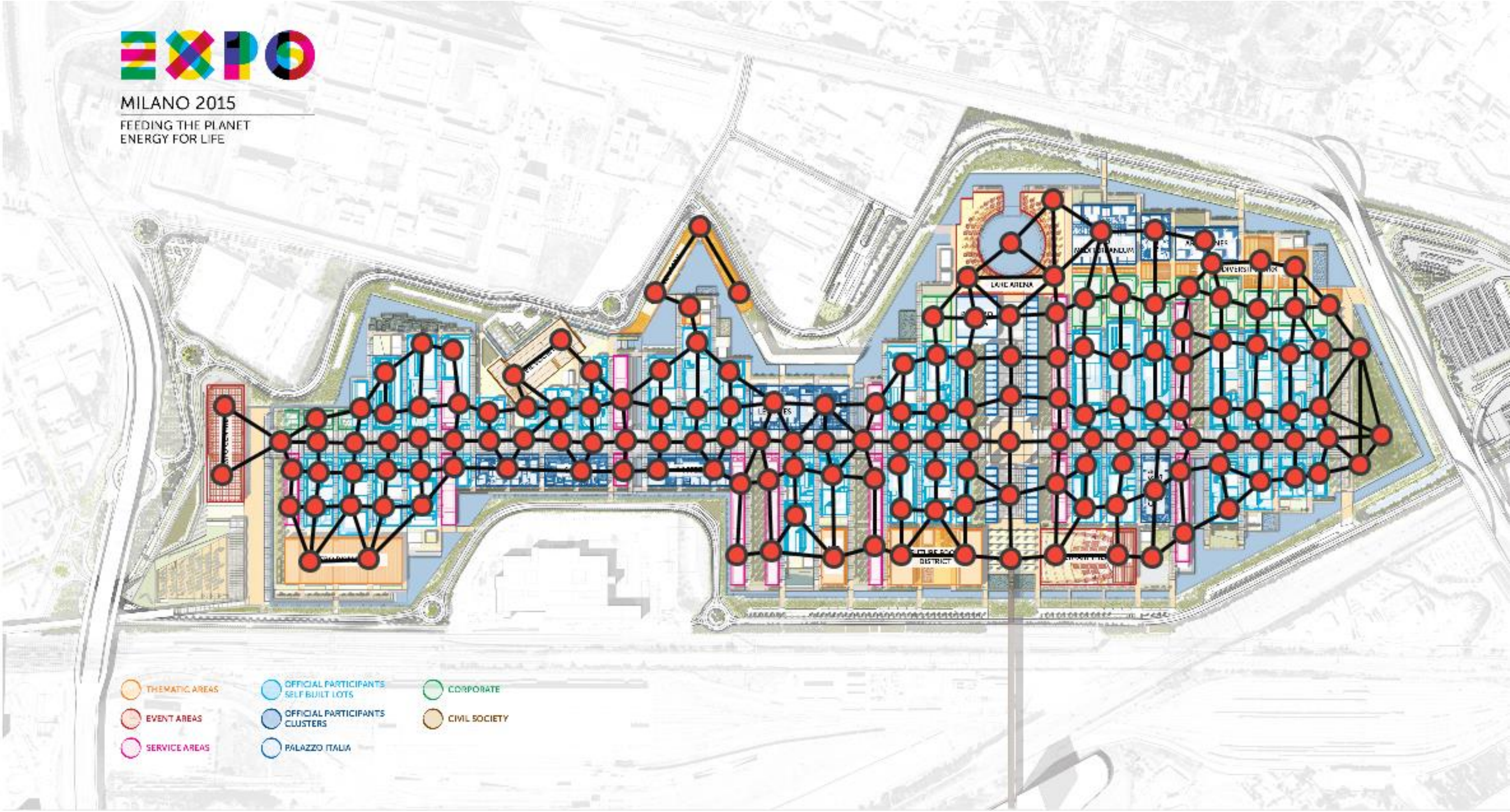
Normal instances: up to 200 targets

- Low edge density
- Spatial locality: distant targets covered by different signals

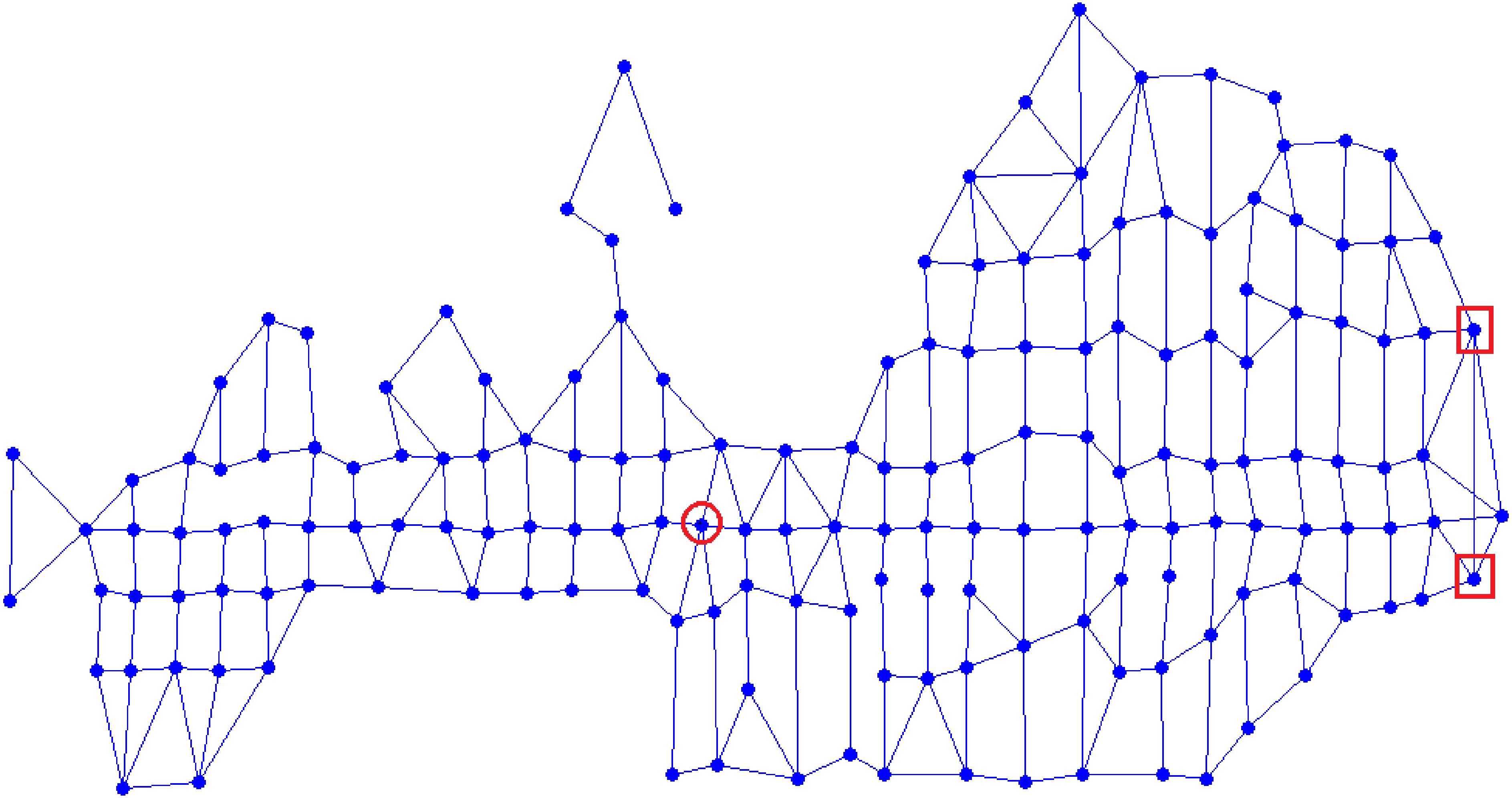
Expo: the setting



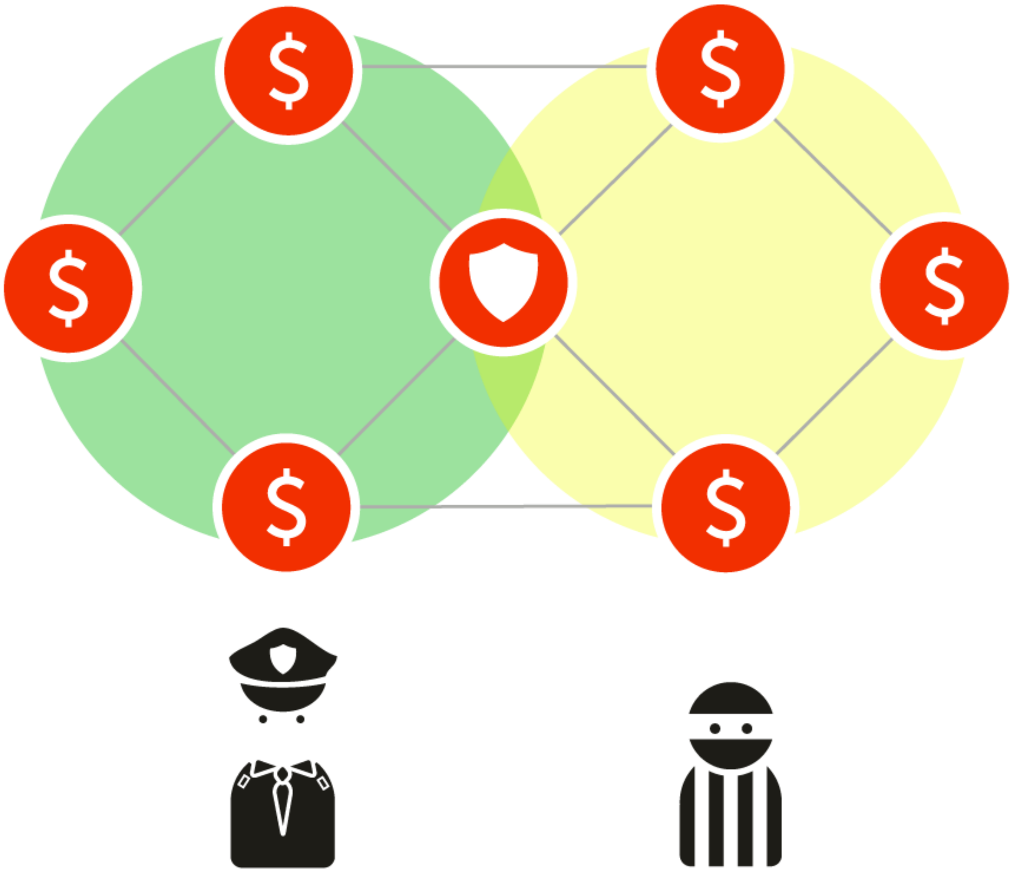
Expo: the graph



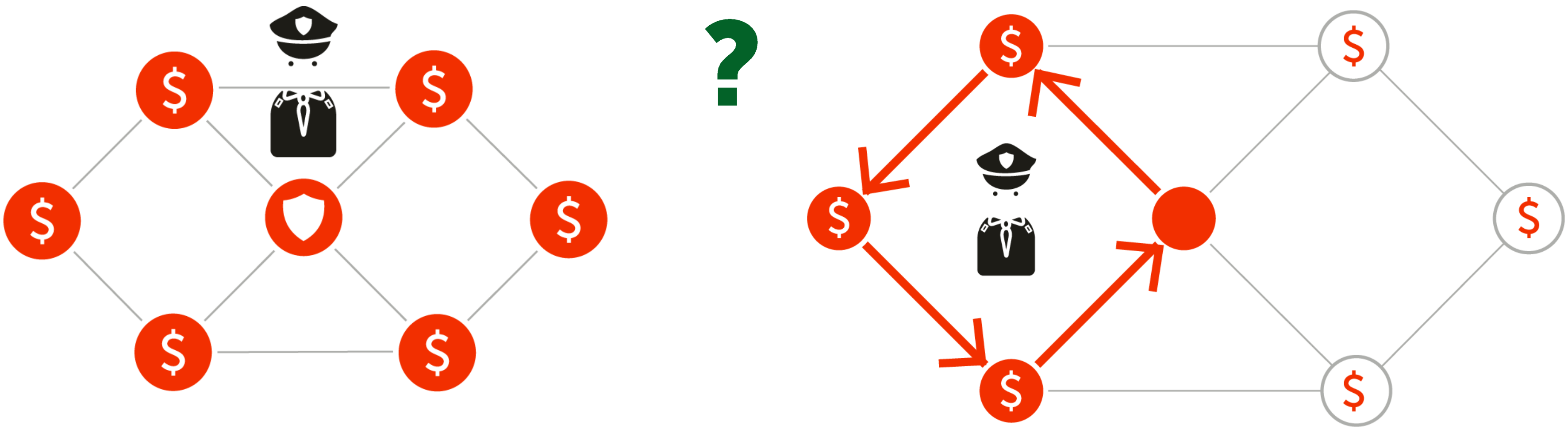
Expo: the solution



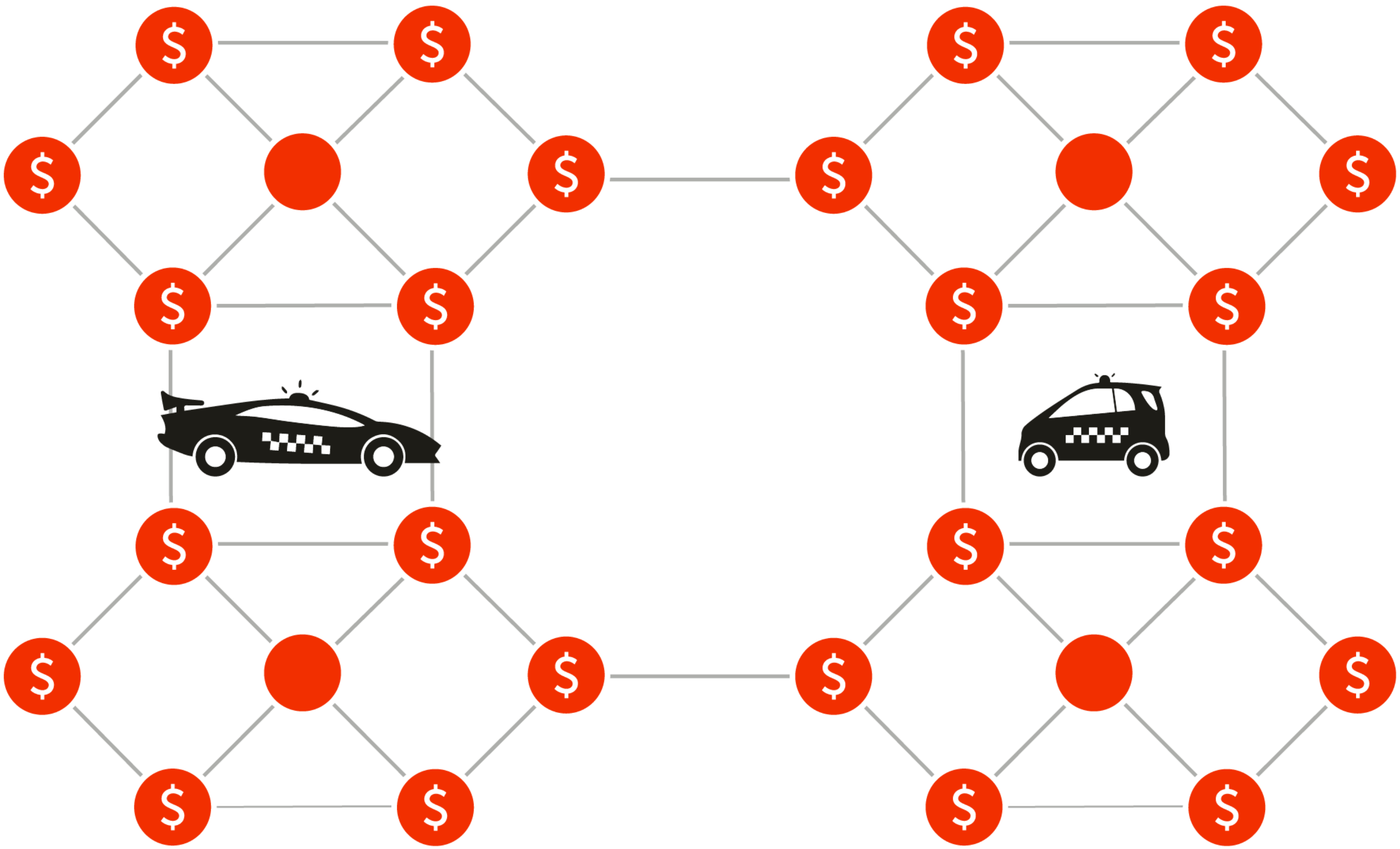
Conclusions



Current research: missed detections



Future research: multi-patrolling



Future research: alarm system deployment

Our Team

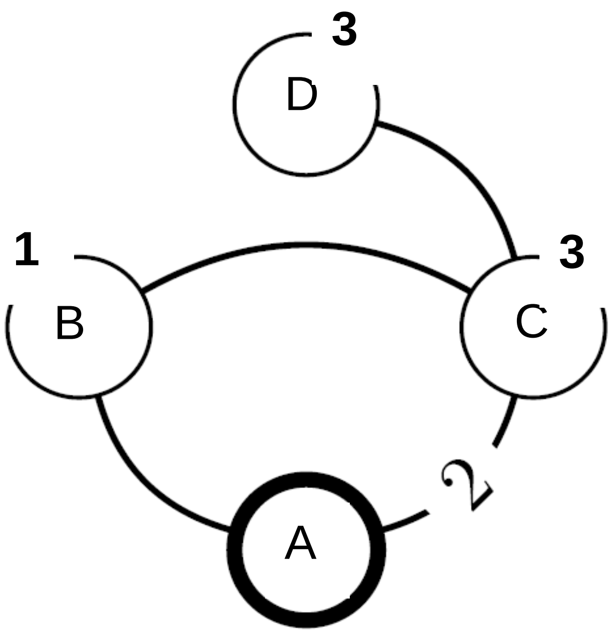
Nicola Gatti

Giuseppe De Nittis

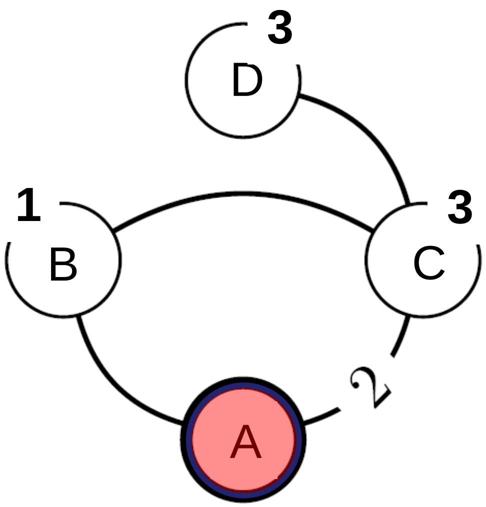
Nicola Basilico

Thank you!

An example



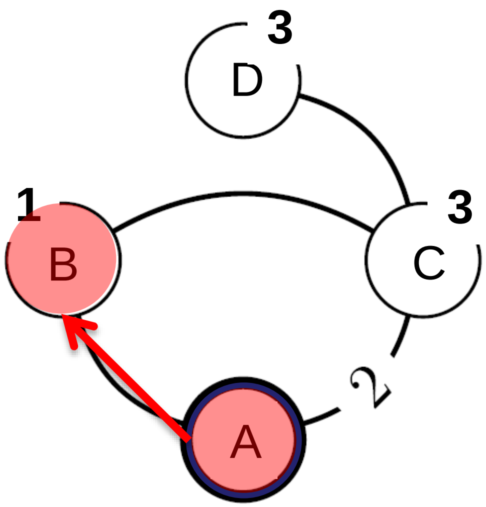
An example



k = 1

‹{A} → A, 0›

An example



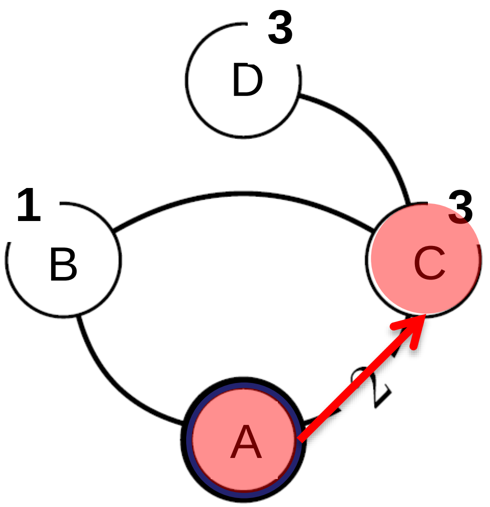
k = 1

‹{A} → A, 0›

k = 2

‹{A,B} → B, 1›

Our Algorithm: an Example



k = 1

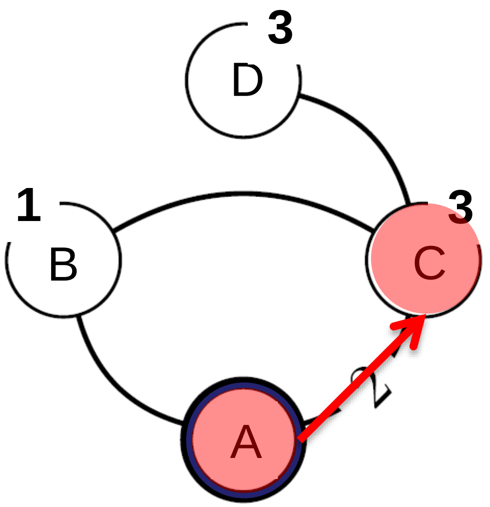
‹{A} → A, 0›

k = 2

‹{A,B} → B, 1›

‹{A,C} → C, 2›

An example



k = 1

‹{A} → A, 0›

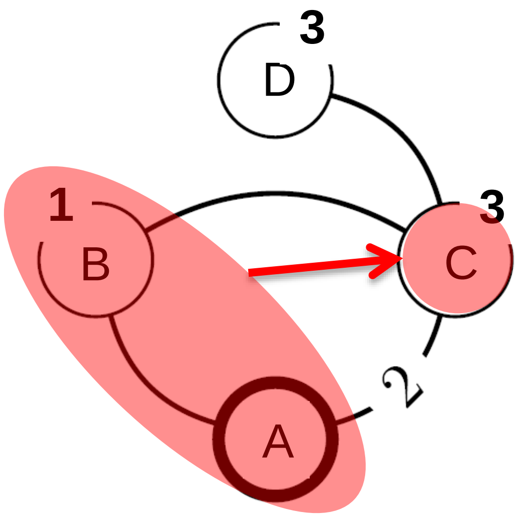
Dominated!

k = 2

‹{A,B} → B, 1›

‹{A,C} → C, 2›

An example



k = 1

~~‹{A} → A, 0›~~

Dominated!

k = 2

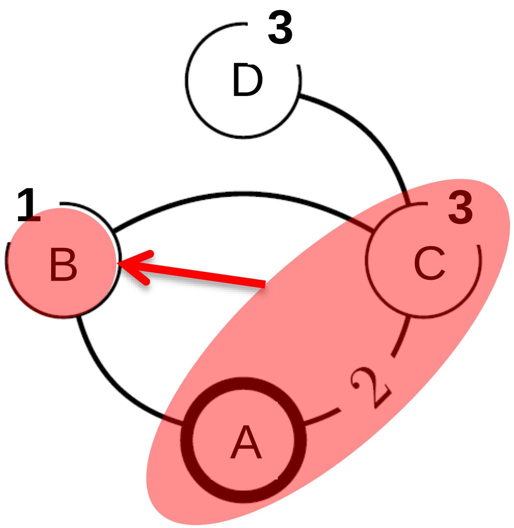
‹{A,B} → B, 1›

‹{A,C} → C, 2›

k = 3

‹{A,B,C} → C, 2›

An example



k = 1

‹{A} → A, 0›

Dominated!

k = 2

‹{A,B} → B, 1›

‹{A,C} → C, 2›

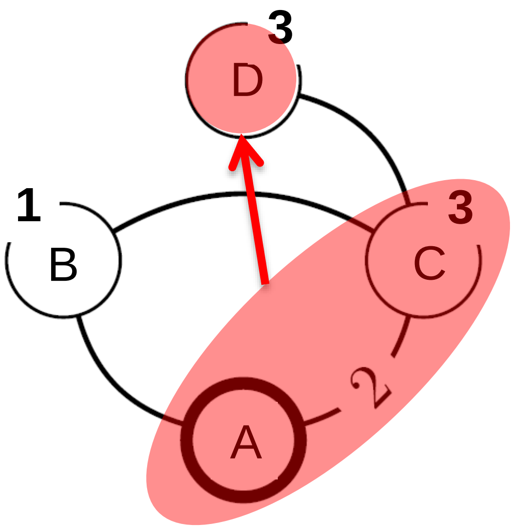
k = 3

‹{A,B,C} → B, 3›

‹{A,B,C} → C, 2›

Unfeasible!

An example



k = 1

‹{A} → A, 0›

Dominated!

k = 2

‹{A,B} → B, 1›

‹{A,C} → C, 2›

k = 3

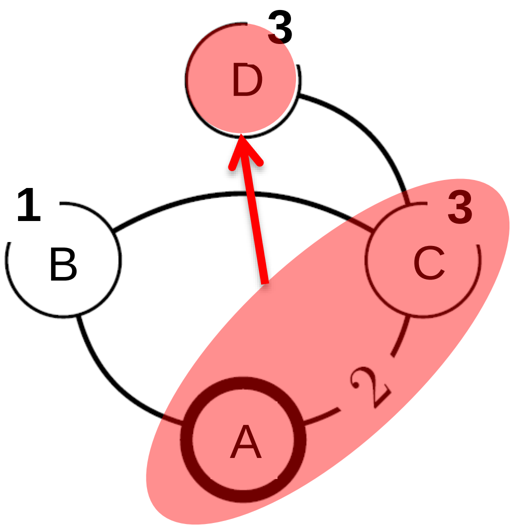
‹{A,B,C} → B, 3›

‹{A,B,C} → C, 2›

‹{A,C,D} → D, 3›

Unfeasible!

An example



k = 1

‹{A} → A, 0›

Dominated!

k = 2

‹{A,B} → B, 1›

‹{A,C} → C, 2›

Dominated!

k = 3

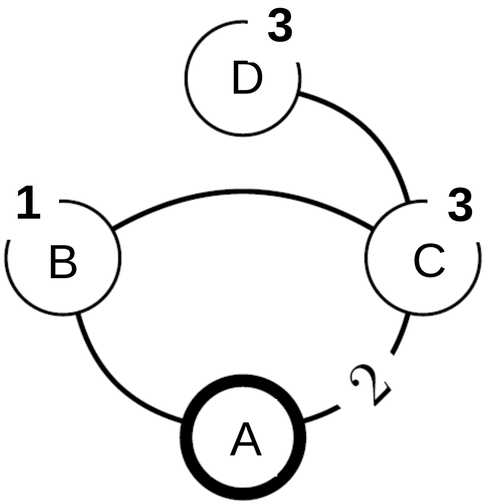
‹{A,B,C} → B, 3›

‹{A,B,C} → C, 2›

‹{A,C,D} → D, 3›

Unfeasible!

An example



k = 1

‹{A} → A, 0›

Dominated!

k = 2

‹{A,B} → B, 1›

‹{A,C} → C, 2›

Dominated!

k = 3

‹{A,B,C} → B, 3›

‹{A,B,C} → C, 2›

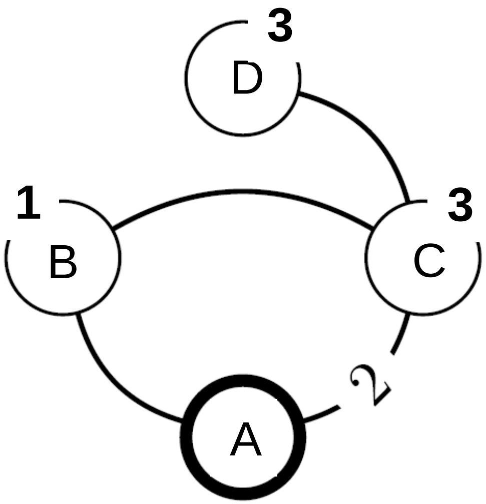
‹{A,C,D} → D, 3›

Unfeasible!

k = 4

‹{A,B,C,D} → D, 3›

An example



k = 1

‹{A} → A, 0›

Dominated!

k = 2

‹{A,B} → B, 1›

‹{A,C} → C, 2›

Dominated!

k = 3

‹{A,B,C} → B, 3›

‹{A,B,C} → C, 2›

‹{A,C,D} → D, 3›

Dominated!

Unfeasible!

k = 4

‹{A,B,C,D} → D, 3›

Computational complexity

The worst-case complexity of the algorithm is

$$O(|T(s)|^2 2^{|T(s)|})$$

since it has to compute proper covering sets up to cardinality $|T(s)|$.

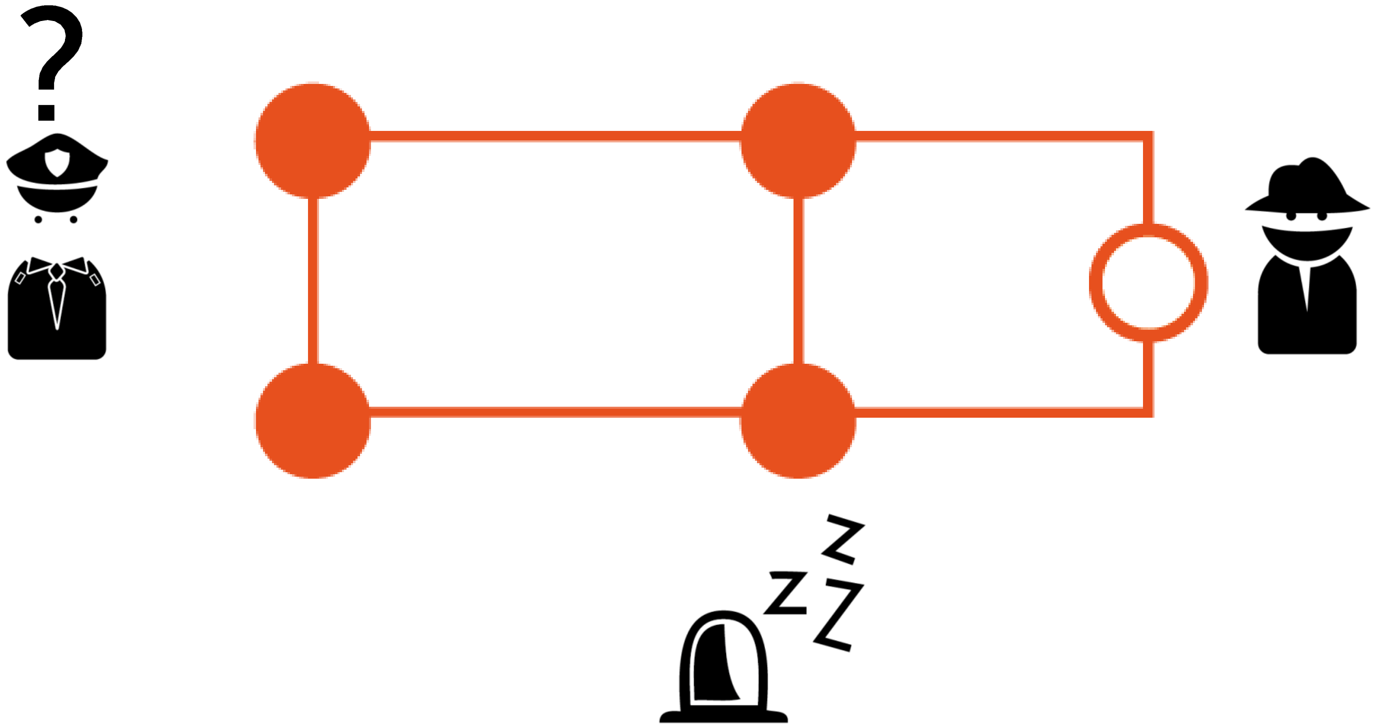
With annotations of dominances and routes generation, the whole algorithm yields a worst-case complexity of $O(|T(s)|^5 2^{|T(s)|})$.

Pseudo-code

Algorithm 1 ComputeCovSets (Basic)

$\forall t \in T, k \in \{2, \ldots, |T|\}, C_t^1 = \{t\}, C_t^k = \emptyset$
$\forall t \in T, c(\{t\}) = \omega^*_{v,t}, c(\emptyset) = \infty$
for all $k \in \{2 \ldots |T|\}$ **do**
  for all $t \in T$ **do**
    for all $Q_t^{k-1} \in C_t^{k-1}$ **do**
      $Q^+ = \{f \in T \setminus Q_t^{k-1} \mid c(Q_t^{k-1}) + \omega^*_{t,f} \leq d(f)\}$
      for all $f \in Q^+$ **do**
        $Q_f^k = Q_t^{k-1} \cup \{f\}$
        $U = Search(Q_f^k, C_f^k)$
        if $c(U) > c(Q_t^{k-1}) + \omega^*_{t,f}$ **then**
          $C_f^k = C_f^k \cup \{Q_f^k\}$
          $c(Q_f^k) = c(Q_t^{k-1}) + \omega^*_{t,f}$
        end if
      end for
    end for
  end for
end for

Missed detections



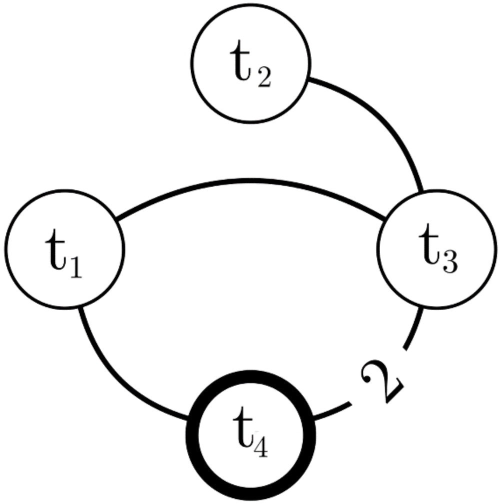
v* is the best placement
u* is the second best placement

$$(1 - \alpha)(1 - g_{v*}) > 1 - g_{u*}$$

Missed detections

$v^* = t_4$
$u^* = t_2$

v^* is the best placement for $\alpha \leq 0.25$

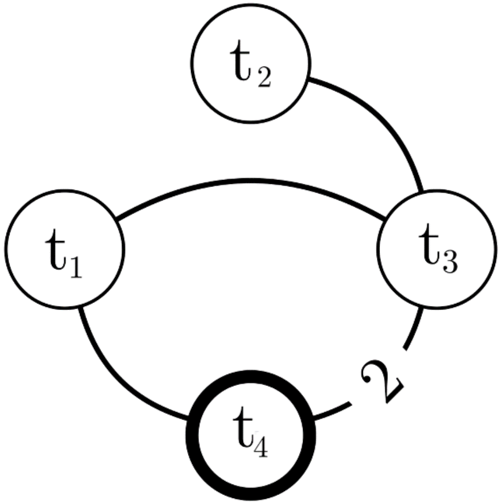


t	$\pi(t)$	d(t)	$p(s_1 \mid t)$
t_1	0.5	1	1.0
t_2	0.5	3	1.0
t_3	0.5	2	1.0
t_4	0.5	2	1.0

Missed detections

$v^* = t_4$
$u^* = t_2$

v^* is the best placement for $\alpha \leq 0.50$



t	$\pi(t)$	d(t)	$p(s_1\|t)$
t_1	1.0	1	1.0
t_2	1.0	3	1.0
t_3	1.0	2	1.0
t_4	1.0	2	1.0

k-SRG-v is NP-hard

We reduce from Hamiltonian Path.
Given an instance of HP, $G_H = (V_H, E_H)$, we build a k-SRG-v instance as follows:

- $V = V_H \cup \{v\}$;
- $E = E_H \cup \{(v,h), h \text{ in } V_H\}$, $w_{i,j} = 1$;
- $T = V_H$, $d(t) = |V_H|$, $\pi(t) = 1$;
- $S = \{s\}$, $p(s|t) = 1$;
- $k = 0$.

If $g_v = 0$, then T must be a covering set that admits at least one covering route r, which visits every node exactly one time.
Since $T = V_H$, $g_v \leq 0$ if and only if G_H admits a Hamiltonian path.

Computing covering sets

Definition: The decision problem COV-SET is defined as:

INSTANCE: an instance of SRG-v with a target set T

QUESTION: is T a covering set for some covering route r?

Theorem: COV-SET is NP-complete.

Security games around the World