

# On the Freeze-Tag Problem

**Cyril Gavoille** joint work with  
N. Bonichon, A. Casteigts, N. Hanusse



GT Graph & Optimization  
LaBRI – Talence  
Avril 7th, 2023



# Context

A robot in charge of awaking a team of asleep robots.

## Subject to:

- awaking by contacts (need a move to meet)
- any awaked robot can help to awake others
- robots lie and move in the Euclidean plane
- constant velocity moves

## Goal:

- to minimize the **time** to awake all the robots
- with a good schedule ... [\[demo\]](#)

# Example (basic)

(1/3)

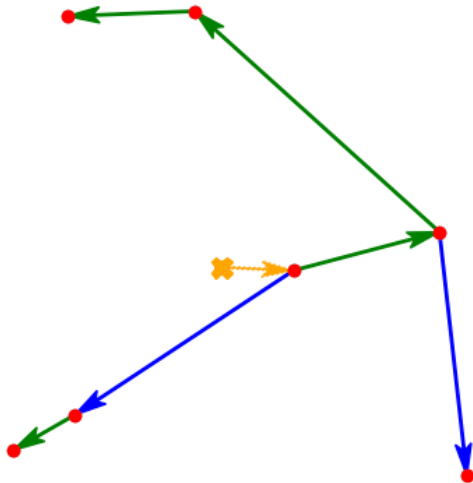
#pts = 8  
diam = 1.890  
ecc = 1.000



# Example (basic)

(1/3)

optimal = 2.110  
depth = 2.110  
#pts = 8  
diam = 1.890  
ecc = 1.000  
time = 0.009s



# Example (Jupiter)

(2/3)

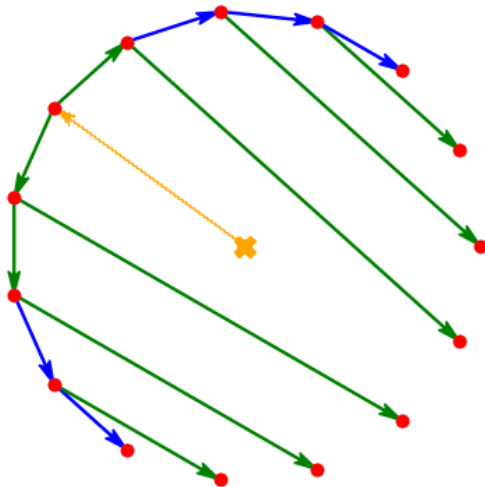
#pts = 16  
diam = 1.989  
ecc = 1.000



# Example (Jupiter)

(2/3)

optimal = 3.318  
depth = 3.318  
#pts = 16  
diam = 1.989  
ecc = 1.000  
time = 207.710s





# Example (convex)

(3/3)



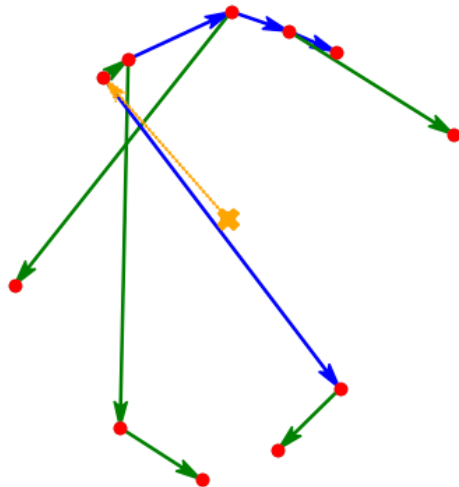
convex pts  
#pts = 12  
diam = 1.783  
ecc = 1.000  
time = 1.372s  
seed = 3906



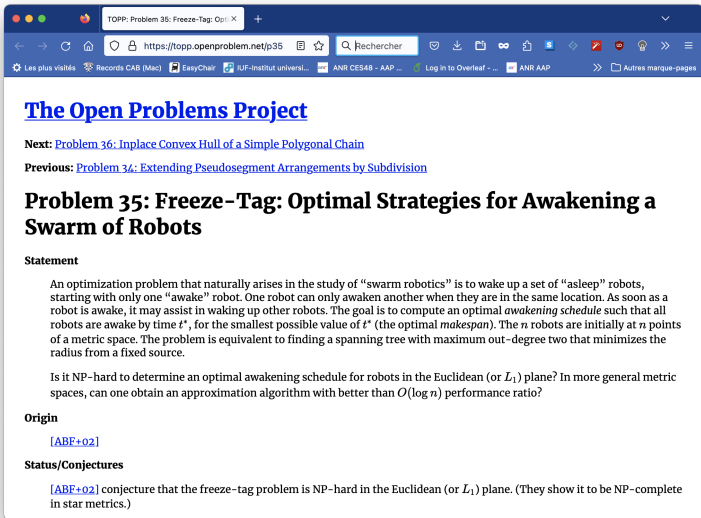
# Example (convex)

(3/3)

optimal = 2.594  
depth = 2.594  
#pts = 12  
diam = 1.783  
ecc = 1.000  
seed = 3906



<https://topp.openproblem.net/p35>



The screenshot shows a web browser window with the address bar containing <https://topp.openproblem.net/p35>. The page title is "TOPP: Problem 35: Freeze-Tag: Optimal Strategies for Awakening a Swarm of Robots". The browser's address bar also shows a search engine "Rechercher". The page content includes a navigation bar with links to "Next: Problem 36: Inplace Convex Hull of a Simple Polygonal Chain" and "Previous: Problem 34: Extending Pseudosegment Arrangements by Subdivision". The main heading is "Problem 35: Freeze-Tag: Optimal Strategies for Awakening a Swarm of Robots". Below this is the "Statement" section, which describes an optimization problem in swarm robotics. The "Origin" section is linked to "[ABE+02]". The "Status/Conjectures" section is linked to "[ABE+02]" and discusses the complexity of the problem.

## The Open Problems Project

Next: [Problem 36: Inplace Convex Hull of a Simple Polygonal Chain](#)

Previous: [Problem 34: Extending Pseudosegment Arrangements by Subdivision](#)

### Problem 35: Freeze-Tag: Optimal Strategies for Awakening a Swarm of Robots

#### Statement

An optimization problem that naturally arises in the study of “swarm robotics” is to wake up a set of “asleep” robots, starting with only one “awake” robot. One robot can only awaken another when they are in the same location. As soon as a robot is awake, it may assist in waking up other robots. The goal is to compute an optimal *awakening schedule* such that all robots are awake by time  $t^*$ , for the smallest possible value of  $t^*$  (the optimal *makespan*). The  $n$  robots are initially at  $n$  points of a metric space. The problem is equivalent to finding a spanning tree with maximum out-degree two that minimizes the radius from a fixed source.

Is it NP-hard to determine an optimal awakening schedule for robots in the Euclidean (or  $L_1$ ) plane? In more general metric spaces, can one obtain an approximation algorithm with better than  $O(\log n)$  performance ratio?

#### Origin

[\[ABE+02\]](#)

#### Status/Conjectures

[\[ABE+02\]](#) conjecture that the freeze-tag problem is NP-hard in the Euclidean (or  $L_1$ ) plane. (They show it to be NP-complete in star metrics.)

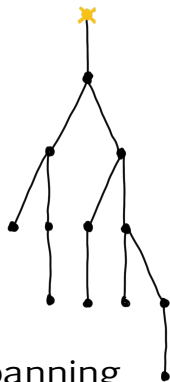
Master student projects at U. Bordeaux [\[youtube\]](#)

# Problem Statement

## FREEZE-TAG PROBLEM (basic)

**Input:** a source  $s \in \mathbb{R}^2$   
a set  $P \subseteq \mathbb{R}^2$  of  $n$  points

**Output:** a *wake-up tree* for  $(s, P)$  of minimum depth w.r.t.  $\ell_2$ -norm



A **wake-up tree** for  $(s, P)$  is a binary tree spanning  $\{s\} \cup P$  of root  $s$  which has at most 1 child.

**Variants** (space for moves):  $\ell_p$ -norm, metric-spaces, weighted graphs, non-metric spaces, ...,  $\mathbb{R}^3$ , ...

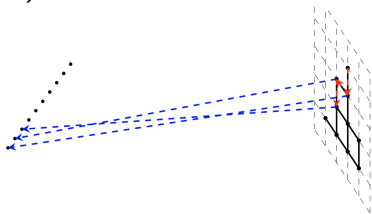
# Known Results: Hardness (1/3)

[ABF+06] NP-Hard for moves restricted to weighted star graphs, from NUMERICAL 3D MATCHING, and for unweighted trees, from 3-PARTITION (asleep robots at leaves)

# Known Results: Hardness (1/3)

[ABF+06] NP-Hard for moves restricted to weighted star graphs, from NUMERICAL 3D MATCHING, and for unweighted trees, from 3-PARTITION (asleep robots at leaves)

[AY16,J17] NP-Hard for  $(\mathbb{R}^3, \ell_2)$ , from HAMILTONIAN PATH IN 2D SUBGRID

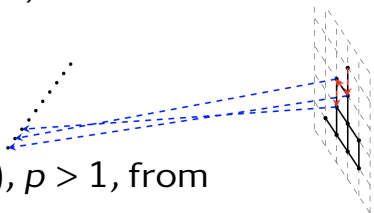


# Known Results: Hardness (1/3)

[ABF+06] NP-Hard for moves restricted to weighted star graphs, from NUMERICAL 3D MATCHING, and for unweighted trees, from 3-PARTITION (asleep robots at leaves)

[AY16,J17] NP-Hard for  $(\mathbb{R}^3, \ell_2)$ , from HAMILTONIAN PATH IN 2D SUBGRID

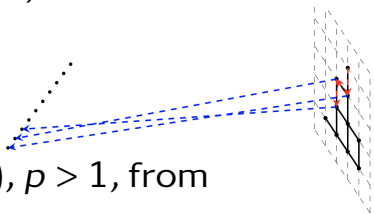
[DR17] NP-Hard for  $(\mathbb{R}^3, \ell_p)$ ,  $p > 1$ , from DOMINATING SET



# Known Results: Hardness (1/3)

[ABF+06] NP-Hard for moves restricted to weighted star graphs, from NUMERICAL 3D MATCHING, and for unweighted trees, from 3-PARTITION (asleep robots at leaves)

[AY16,J17] NP-Hard for  $(\mathbb{R}^3, \ell_2)$ , from HAMILTONIAN PATH IN 2D SUBGRID



[DR17] NP-Hard for  $(\mathbb{R}^3, \ell_p)$ ,  $p > 1$ , from DOMINATING SET

[AAY17] NP-Hard for  $(\mathbb{R}^2, \ell_2)$ , from MONOTONE 3SAT

# Known Results: Approximation (2/3)

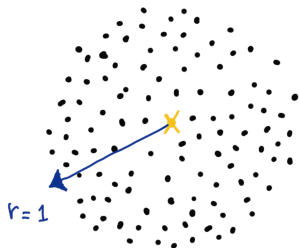
[ABF+06]  $O(\log n)$ -approximation for locally-bounded weighted graphs (bounded aspect ratio for incident edges)





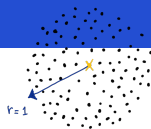
# Known Results: Upper Bound (3/3)

[YBMK15] For  $(\mathbb{R}^2, \ell_2)$  and unit source radius,  
 $\exists$  wake-up tree of depth  $< 10.1$   
computable in  $O(n)$  time

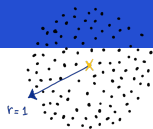


# Contributions

For  $(\mathbb{R}^2, \ell_2)$  and unit source radius



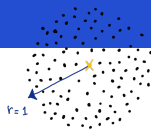
# Contributions



For  $(\mathbb{R}^2, \ell_2)$  and unit source radius

- $\forall P$  in convex position,  $\exists$  wake-up tree of depth  $\leq 1 + 2\sqrt{2} \approx 3.8$ , achieved for  $n = |P| = 4$

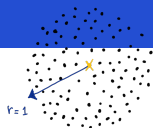
# Contributions



For  $(\mathbb{R}^2, \ell_2)$  and unit source radius

- $\forall P$  in convex position,  $\exists$  wake-up tree of depth  $\leq 1 + 2\sqrt{2} \approx 3.8$ , achieved for  $n = |P| = 4$
- $\forall P, \exists$  wake-up tree of depth  $\leq 3 + O(1/\sqrt{n})$

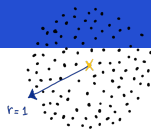
# Contributions



For  $(\mathbb{R}^2, \ell_2)$  and unit source radius

- $\forall P$  in convex position,  $\exists$  wake-up tree of depth  $\leq 1 + 2\sqrt{2} \approx 3.8$ , achieved for  $n = |P| = 4$
- $\forall P$ ,  $\exists$  wake-up tree of depth  $\leq 3 + O(1/\sqrt{n})$
- $\exists P$ ,  $\forall$  wake-up tree has depth  $\geq 3 + \Omega(1/n^{1/3})$

# Contributions



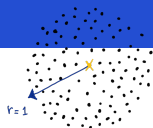
For  $(\mathbb{R}^2, \ell_2)$  and unit source radius

- $\forall P$  in convex position,  $\exists$  wake-up tree of depth  $\leq 1 + 2\sqrt{2} \approx 3.8$ , achieved for  $n = |P| = 4$
- $\forall P$ ,  $\exists$  wake-up tree of depth  $\leq 3 + O(1/\sqrt{n})$
- $\exists P$ ,  $\forall$  wake-up tree has depth  $\geq 3 + \Omega(1/n^{1/3})$

For  $(\mathbb{R}^2, \ell_1)$  and unit source radius

- $\forall P$ ,  $\exists$  wake-up tree of depth  $\leq 5$  (optimal)

# Contributions



For  $(\mathbb{R}^2, \ell_2)$  and unit source radius

- $\forall P$  in convex position,  $\exists$  wake-up tree of depth  $\leq 1 + 2\sqrt{2} \approx 3.8$ , achieved for  $n = |P| = 4$
- $\forall P$ ,  $\exists$  wake-up tree of depth  $\leq 3 + O(1/\sqrt{n})$
- $\exists P$ ,  $\forall$  wake-up tree has depth  $\geq 3 + \Omega(1/n^{1/3})$

For  $(\mathbb{R}^2, \ell_1)$  and unit source radius

- $\forall P$ ,  $\exists$  wake-up tree of depth  $\leq 5$  (optimal)

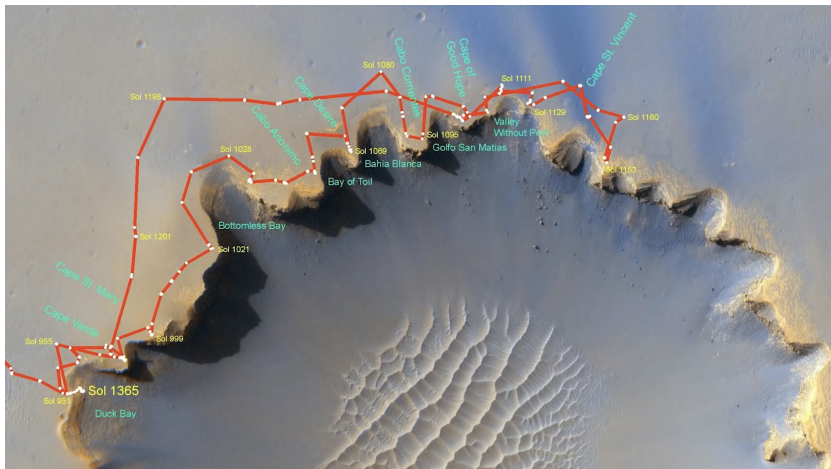
Exact algorithm running in time  $3^n \cdot n^{O(1)}$

(holds for any non-metric space), experiments,  
heuristics, ...



# Brute-Force Algorithms

Looks like to TSP (cycle or path vs. binary tree)



$n!c^n$  outputs (= wake-up trees) to explore !

# Dynamic Programming

FOR RELEASE:  
A. M.'s, Thursday  
January 2, 1964

FROM: International Business Machines Corp.  
Data Processing Division  
112 East Post Road  
White Plains, New York

Bert Reisman  
914 White Plains 9-1900

WHITE PLAINS, N. Y., Jan. 2 . . . . IBM mathematicians (left to right) Michael Held, Richard Shreshian and Richard M. Karp review the manual describing a new computer program which provides business and industry with a practical scientific method for handling a wide variety of complex scheduling tasks. The program, available to users of the IBM 7090 and 7094 data processing systems, consists of a set of 4,500 instructions which tell the computer what to do with data fed into it. It grew out of the trio's efforts to find solutions for a classic mathematical problem -- the "Traveling Salesman" problem -- which has long defied solution by man, or by the fastest computers he uses.



[Held-Karp'64]  $O(n^2 2^n)$

Best known complexity for non-metric TSP

# Recurrence & Memorization

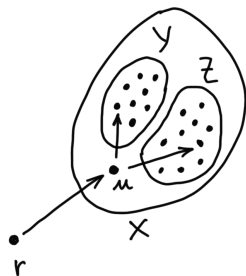
$\text{opt}(r, X, b)$  := optimal time to awake all asleep robots of  $X \subseteq P$  from  $r \notin X$  with  $b \in \{1, 2\}$  awaked robots at  $r$

$\text{opt}(s, P, 1)$ ?

# Recurrence & Memorization

$\text{opt}(r, X, b) :=$  optimal time to awake all asleep robots of  $X \subseteq P$  from  $r \notin X$  with  $b \in \{1, 2\}$  awaked robots at  $r$

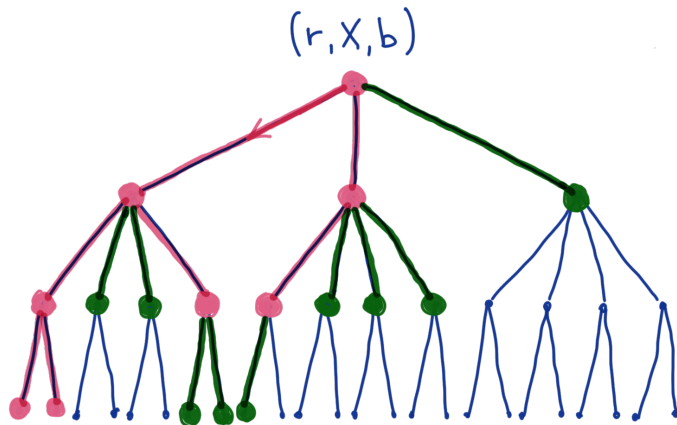
$\text{opt}(s, P, 1)?$



$\text{opt}(r, X, 1) = 0$  if  $|X| = 0$

$\text{opt}(r, X, 1) = \min_{u \in X} \{ \text{dist}(r, u) + \text{opt}(u, X \setminus \{u\}, 2) \}$

$\text{opt}(u, X, 2) = \min_{X=Y \dot{\cup} Z} \max \{ \text{opt}(u, Y, 1), \text{opt}(u, Z, 1) \}$



[ #nodes  $\gg n! \cdot 2^n$  ]

$$\text{TIME}(r, X, b) = \# \text{visit}(r, X, b) \times \text{lookup}(r, X, b) \\ + \#\{(r, X, b)\} \times \text{cost}(r, X, b)$$

$$\begin{aligned} \text{TIME}(r, X, b) = & \# \text{visit}(r, X, b) \times \text{lookup}(r, X, b) \\ & + \# \{(r, X, b)\} \times \text{cost}(r, X, b) \end{aligned}$$

$$\text{TIME}(r, X, b) = \# \text{visit}(r, X, b) \times \text{lookup}(r, X, b) \\ + \# \{(r, X, b)\} \times \text{cost}(r, X, b)$$

$$\# \{(r, X, b)\} = (1 + n) \times \binom{n}{x} \times 2 \qquad x := |X|$$

$$\text{TIME}(r, X, b) = \# \text{visit}(r, X, b) \times \text{lookup}(r, X, b) \\ + \# \{(r, X, b)\} \times \text{cost}(r, X, b)$$

$$\# \{(r, X, b)\} = (1 + n) \times \binom{n}{x} \times 2 \\ \# \text{visit} < \# \{(r, X, b)\} \times \Delta$$

$$x := |X| \\ \Delta = 2^x$$



$$\text{TIME}(r, X, b) = \# \text{visit}(r, X, b) \times \text{lookup}(r, X, b) \\ + \# \{(r, X, b)\} \times \text{cost}(r, X, b)$$

$$\# \{(r, X, b)\} = (1 + n) \times \binom{n}{x} \times 2$$

$$x := |X|$$

$$\# \text{visit} < \# \{(r, X, b)\} \times \Delta$$

$$\Delta = 2^x$$

$$\text{lookup} = O(\log(\# \{(r, X, b)\})) = O(n)$$

$$\text{TIME}(r, X, b) = \# \text{visit}(r, X, b) \times \text{lookup}(r, X, b) \\ + \# \{(r, X, b)\} \times \text{cost}(r, X, b)$$

$$\# \{(r, X, b)\} = (1 + n) \times \binom{n}{x} \times 2$$

$$x := |X|$$

$$\# \text{visit} < \# \{(r, X, b)\} \times \Delta$$

$$\Delta = 2^x$$

$$\text{lookup} = O(\log(\# \{(r, X, b)\})) = O(n)$$

$$\text{cost} = O(\text{lookup} + x \cdot 2^x)$$

$$\text{TIME}(r, X, b) = \# \text{visit}(r, X, b) \times \text{lookup}(r, X, b) \\ + \# \{(r, X, b)\} \times \text{cost}(r, X, b)$$

$$\# \{(r, X, b)\} = (1 + n) \times \binom{n}{x} \times 2$$

$$x := |X|$$

$$\# \text{visit} < \# \{(r, X, b)\} \times \Delta$$

$$\Delta = 2^x$$

$$\text{lookup} = O(\log(\# \{(r, X, b)\})) = O(n)$$

$$\text{cost} = O(\text{lookup} + x \cdot 2^x)$$

$$\max_{X \subseteq P} \left\{ 2^x \cdot \binom{n}{x} \right\} = 3^n / \Theta(\sqrt{n})$$

$$\text{TIME}(r, X, b) = \# \text{visit}(r, X, b) \times \text{lookup}(r, X, b) \\ + \# \{(r, X, b)\} \times \text{cost}(r, X, b)$$

$$\# \{(r, X, b)\} = (1 + n) \times \binom{n}{x} \times 2$$

$$x := |X|$$

$$\# \text{visit} < \# \{(r, X, b)\} \times \Delta$$

$$\Delta = 2^x$$

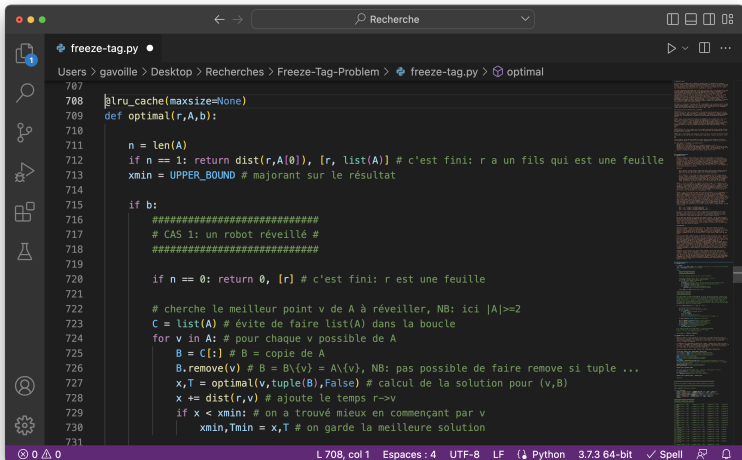
$$\text{lookup} = O(\log(\# \{(r, X, b)\})) = O(n)$$

$$\text{cost} = O(\text{lookup} + x \cdot 2^x)$$

$$\max_{X \subseteq P} \left\{ 2^x \cdot \binom{n}{x} \right\} = 3^n / \Theta(\sqrt{n})$$

$$\Rightarrow \boxed{\text{TIME}(s, P, 1) = 3^n \cdot O(n^{3/2})}$$

# Python



```
707
708 |lru_cache(maxsize=None)
709 def optimal(r,A,b):
710
711     n = len(A)
712     if n == 1: return dist(r,A[0]), [r, list(A)] # c'est fini: r a un fils qui est une feuille
713     xmin = UPPER_BOUND # majorant sur le resultat
714
715     if b:
716         #####
717         # CAS 1: un robot reveillé #
718         #####
719
720     if n == 0: return 0, [r] # c'est fini: r est une feuille
721
722     # cherche le meilleur point v de A à reveiller, NB: ici |A|>=2
723     C = list(A) # évite de faire list(A) dans la boucle
724     for v in A: # pour chaque v possible de A
725         B = C[:] # B = copie de A
726         B.remove(v) # B = B \ {v} = A \ {v}, NB: pas possible de faire remove si tuple ...
727         x,T = optimal(v,tuple(B),False) # calcul de la solution pour (v,B)
728         x += dist(r,v) # ajoute le temps r->v
729         if x < xmin: # on a trouvé mieux en commençant par v
730             xmin,Tmin = x,T # on garde la meilleure solution
731
```

In practice, 17 points takes 5'30" (Python) vs.  
 $17! \times 10^{-9}s \approx 355,687s \approx 4.1$  days

## Bounds on $\alpha_n$

$\alpha_n :=$  infimum real st. every unit source radius and  $n$  point set has a wake-up tree of depth  $\leq \alpha_n$  for  $(\mathbb{R}^2, \ell_2)$

## Bounds on $\alpha_n$

$\alpha_n :=$  infimum real st. every unit source radius and  $n$  point set has a wake-up tree of depth  $\leq \alpha_n$  for  $(\mathbb{R}^2, \ell_2)$

$$\alpha_0 = 0$$

## Bounds on $\alpha_n$

$\alpha_n :=$  infimum real st. every unit source radius and  $n$  point set has a wake-up tree of depth  $\leq \alpha_n$  for  $(\mathbb{R}^2, \ell_2)$

$$\alpha_0 = 0$$

$$\alpha_1 = 1$$



## Bounds on $\alpha_n$

$\alpha_n :=$  infimum real st. every unit source radius and  $n$  point set has a wake-up tree of depth  $\leq \alpha_n$  for  $(\mathbb{R}^2, \ell_2)$

$$\alpha_0 = 0$$

$$\alpha_1 = 1$$

$$\alpha_2 = 3, \quad \forall n \geq 2, \alpha_n \geq 3$$

[demo]

## Bounds on $\alpha_n$

$\alpha_n :=$  infimum real st. every unit source radius and  $n$  point set has a wake-up tree of depth  $\leq \alpha_n$  for  $(\mathbb{R}^2, \ell_2)$

$$\alpha_0 = 0$$

$$\alpha_1 = 1$$

$$\alpha_2 = 3, \quad \forall n \geq 2, \alpha_n \geq 3$$

$$\alpha_3 = 3$$

[demo]

# Bounds on $\alpha_n$

$\alpha_n$  := infimum real st. every unit source radius and  $n$  point set has a wake-up tree of depth  $\leq \alpha_n$  for  $(\mathbb{R}^2, \ell_2)$

$$\alpha_0 = 0$$

$$\alpha_1 = 1$$

$$\alpha_2 = 3, \quad \forall n \geq 2, \alpha_n \geq 3 \quad [\text{demo}]$$

$$\alpha_3 = 3$$

$$\alpha_4 = 1 + 2\sqrt{2} \approx 3.8 \quad [\text{demo}]$$

# Bounds on $\alpha_n$

$\alpha_n$  := infimum real st. every unit source radius and  $n$  point set has a wake-up tree of depth  $\leq \alpha_n$  for  $(\mathbb{R}^2, \ell_2)$

$$\alpha_0 = 0$$

$$\alpha_1 = 1$$

$$\alpha_2 = 3, \quad \forall n \geq 2, \alpha_n \geq 3 \quad [\text{demo}]$$

$$\alpha_3 = 3$$

$$\alpha_4 = 1 + 2\sqrt{2} \approx 3.8 \quad [\text{demo}]$$

$$\alpha_{5,6,7} \leq \alpha_4$$

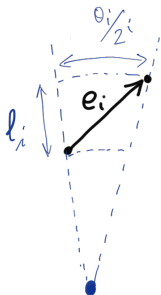
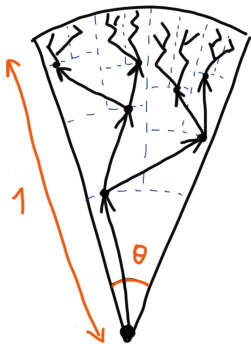
$\vdots$

$$\alpha_n < 5 + 2\sqrt{2} + \sqrt{5} \approx 10.1 \quad \text{from [YBMK15]}$$

# Bounds on $\alpha_n$

## Theorem (warm-up)

$$\forall n \in \mathbb{N}^*, \quad \alpha_n < 3 + 8\pi/\sqrt{n}$$



$$\underbrace{\sum_i l_i}_1 + \underbrace{\sum_i \frac{\theta_i}{2^i}}_{2\theta}$$

The time to awake robots in a slope- $\theta$  cone from its apex is  $\text{cone}(\theta) < 1 + 2\theta$

## Bounds on $\alpha_n$

Theorem (warm-up)

$$\forall n \in \mathbb{N}^*, \quad \alpha_n < 3 + 8\pi/\sqrt{n}$$

$$\text{cone}(\theta) < 1 + 2\theta$$

$$\Rightarrow \alpha_n \leq \text{cone}(2\pi) < 1 + 4\pi < 13.6$$

# Bounds on $\alpha_n$

## Theorem (warm-up)

$$\forall n \in \mathbb{N}^*, \quad \alpha_n < 3 + 8\pi/\sqrt{n}$$

$$\text{cone}(\theta) < 1 + 2\theta$$

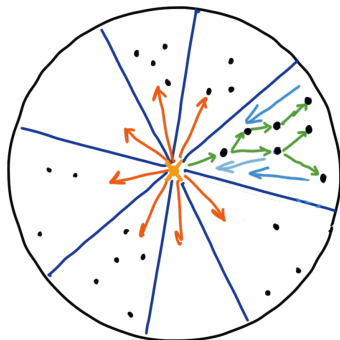
$$\Rightarrow \alpha_n \leq \text{cone}(2\pi) < 1 + 4\pi < 13.6$$

$$\Rightarrow \alpha_n \leq 2 + \text{cone}(\pi) < 3 + 2\pi < 9.3$$

# Bounds on $\alpha_n$

## Theorem (warm-up)

$$\forall n \in \mathbb{N}^*, \quad \alpha_n < 3 + 8\pi/\sqrt{n}$$



- ① split in  $\sqrt{n}$  cones
- ② awake a dense cone
- ③ go back to the source
- ④ awake all cones

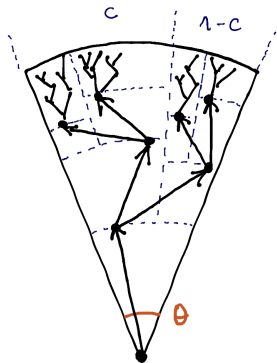
$$\Rightarrow \alpha_n \leq 2\text{cone}(2\pi/\sqrt{n}) + 1 < 3 + 8\pi/\sqrt{n}$$

$$[ \alpha_n < 10.1 \text{ if } n \geq 13 \text{ and } \alpha_n < \alpha_4 \text{ if } n \geq 921 ]$$



# Better Cones!

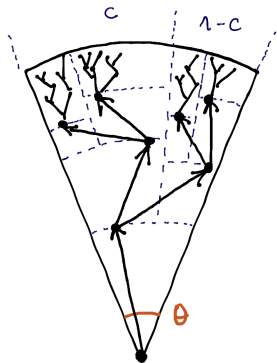
**Strategy.** Minimize zigzags, larger region for the awaked robot, 61% better than 50%!



- Monotone in height:  $\sum_i \ell_i = 1$

# Better Cones!

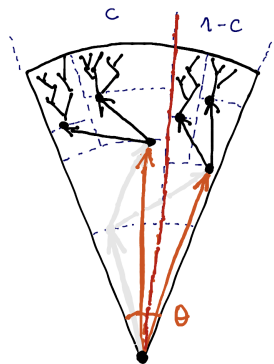
**Strategy.** Minimize zigzags, larger region for the awaked robot, 61% better than 50%!



- Monotone in height:  $\sum_i \ell_i = 1$
- Contribution in  $\theta$  (zigzags):  $f(\theta)$ ?

# Better Cones!

**Strategy.** Minimize zigzags, larger region for the awaked robot, 61% better than 50%!



- Contribution in  $\theta$  (zigzags):

$$c_{\text{opt}}? \quad f(\theta) = \max\{\theta + f((1-c)\theta), c\theta + f(c\theta)\}$$

## Solving $f(\theta)$ ...

$f(\theta) = \alpha\theta$ , for some  $\alpha$

$$f(\theta) = \max\{\theta + f((1-c)\theta), c\theta + f(c\theta)\}$$

$$\alpha\theta = \max\{\theta + \alpha(1-c)\theta, c\theta + \alpha c\theta\}$$

$$c? \quad c \geq 1/\alpha \text{ and } c \geq \alpha/(1+\alpha)$$

$$1/\alpha = \alpha/(1+\alpha) \quad \Leftrightarrow \quad 1+\alpha = \alpha^2$$

$$\Rightarrow \alpha = (1 + \sqrt{5})/2 = \varphi, \quad c_{\text{opt}} = 1/\alpha \approx 61\%$$

### Lemma

$$\text{cone}(\theta) \leq 1 + f(\theta) = 1 + \varphi\theta$$

## Solving $f(\theta)$ ...

$$f(\theta) = \alpha\theta, \text{ for some } \alpha$$

$$f(\theta) = \max\{\theta + f((1-c)\theta), c\theta + f(c\theta)\}$$

$$\alpha\theta = \max\{\theta + \alpha(1-c)\theta, c\theta + \alpha c\theta\}$$

$$c? \quad c \geq 1/\alpha \text{ and } c \geq \alpha/(1+\alpha)$$

$$1/\alpha = \alpha/(1+\alpha) \quad \Leftrightarrow \quad 1+\alpha = \alpha^2$$

$$\Rightarrow \alpha = (1 + \sqrt{5})/2 = \varphi, \quad c_{\text{opt}} = 1/\alpha \approx 61\%$$

### Lemma

$$\text{cone}(\theta) \leq 1 + f(\theta) = 1 + \varphi\theta$$

$$\Rightarrow \alpha_n \leq 2 + \text{cone}(\pi) \leq 3 + \varphi\pi < 8.1$$

# New bounds on $\alpha_n$

## Theorem

$$\forall n \in \mathbb{N}^*, \quad \alpha_n \leq 3 + 4\varphi\pi/\sqrt{n}$$

with some refinements,  $\alpha_n \leq 1 + 2\sqrt{2}$  if  $n \geq 285$ .

# Other Shapes

$\text{rect}(w, h)$ : time to awake a  $w \times h$  rectangle

## Lemma

$$\text{rect}(w, h) \leq w + (1 + \varphi)h$$

# Other Shapes

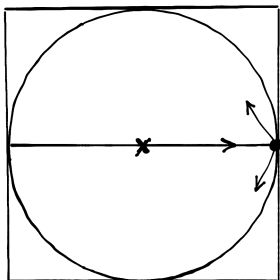
$\text{rect}(w, h)$ : time to awake a  $w \times h$  rectangle

## Lemma

$$\text{rect}(w, h) \leq w + (1 + \varphi)h$$

## Theorem

$$\forall n \in \mathbb{N}^*, \quad \alpha_n \leq 1 + \text{rect}(2, 1) \leq 4 + \varphi < 5.7$$





# Other Shapes

$\text{rect}(w, h)$ : time to awake a  $w \times h$  rectangle

## Lemma

$$\text{rect}(w, h) \leq w + (1 + \varphi)h$$

## Theorem

$$\forall n \in \mathbb{N}^*, \quad \alpha_n \leq 1 + \text{rect}(2, 1) \leq 4 + \varphi < 5.7$$

... and more generally for  $\ell_p$ -norms

## Theorem

$$\forall n \in \mathbb{N}^*, \forall p \geq 1, \quad \alpha_{n,p} \leq 4 + \varphi + \lambda(p) < 6.2$$

where  $\lambda(p) \in [0, \frac{1}{2}]$  and  $\lambda(2) = 0$

# Open Problems

- $\alpha_n \leq \alpha_4$  for every  $n$ ?

# Open Problems

- $\alpha_n \leq \alpha_4$  for every  $n$ ?
- $\alpha_{n,p} \leq \alpha_{4,p} = 1 + 2^{1+\max\{1/p, 1-1/p\}}$  for all  $n$  and  $p$ ?

# Open Problems

- $\alpha_n \leq \alpha_4$  for every  $n$ ?
- $\alpha_{n,p} \leq \alpha_{4,p} = 1 + 2^{1+\max\{1/p, 1-1/p\}}$  for all  $n$  and  $p$ ?
- Polynomial time if  $P$  is convex?

# Open Problems

- $\alpha_n \leq \alpha_4$  for every  $n$ ?
- $\alpha_{n,p} \leq \alpha_{4,p} = 1 + 2^{1+\max\{1/p, 1-1/p\}}$  for all  $n$  and  $p$ ?
- Polynomial time if  $P$  is convex?

THAT'S THE END

