

Distributed Freeze-Tag: How to efficiently wake up a swarm of sleeping robots

Cyril Gavoille, Nicolas Hanusse, Gabriel Le Boudier, Taïssir Marcé

PODC '25 – Huatulco

June 19, 2025

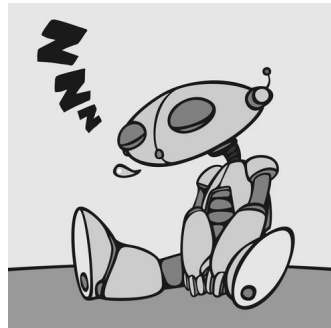


Context: Energy Saving

Sleeping Model

To save energy:

- Put devices into a sleeping mode,
- Schedule awakening period.



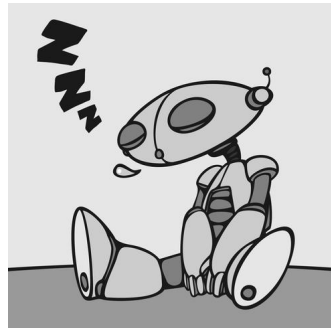
Context: Energy Saving

Sleeping Model

To save energy:

- Put devices into a sleeping mode,
- Schedule awakening period.

⇒ Awakening schedule pre-computed



Context: Energy Saving

Sleeping Model

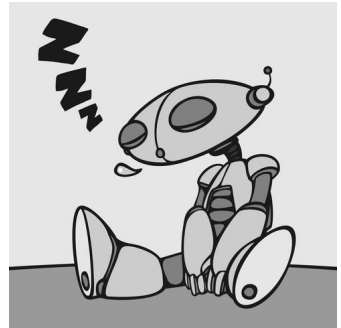
To save energy:

- Put devices into a sleeping mode,
- Schedule awakening period.

⇒ Awakening schedule pre-computed

Long-Time Sleeping

Devices are put into sleep for an unknown duration: until one event requires them to be activated.



Context: Energy Saving

Sleeping Model

To save energy:

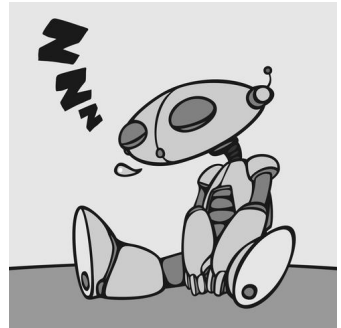
- Put devices into a sleeping mode,
- Schedule awakening period.

⇒ Awakening schedule pre-computed

Long-Time Sleeping

Devices are put into sleep for an unknown duration: until one event requires them to be activated.

⇒ Devices must be manually awaken



The Freeze-Tag Problem [Arkin et al. SODA '02]

Input:

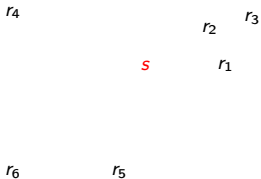
- n sleeping robots in the plane

 r_4 r_2 r_3 r_1 r_6 r_5

The Freeze-Tag Problem [Arkin et al. SODA '02]

Input:

- n sleeping robots in the plane
- robot s , initially awake



The Freeze-Tag Problem [Arkin et al. SODA '02]

Input:

- n sleeping robots in the plane
- robot s , initially awake

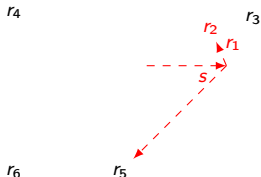


→ awake robots can move and
wake up sleeping robots

The Freeze-Tag Problem [Arkin et al. SODA '02]

Input:

- n sleeping robots in the plane
- robot s , initially awake

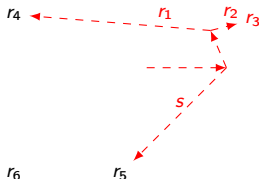


→ awake robots can move and wake up sleeping robots

The Freeze-Tag Problem [Arkin et al. SODA '02]

Input:

- n sleeping robots in the plane
- robot s , initially awake

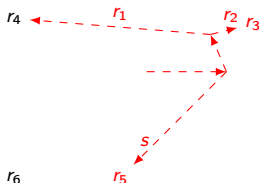


→ awake robots can move and wake up sleeping robots

The Freeze-Tag Problem [Arkin et al. SODA '02]

Input:

- n sleeping robots in the plane
- robot s , initially awake

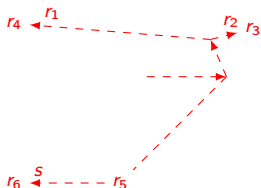


→ awake robots can move and wake up sleeping robots

The Freeze-Tag Problem [Arkin et al. SODA '02]

Input:

- n sleeping robots in the plane
- robot s , initially awake

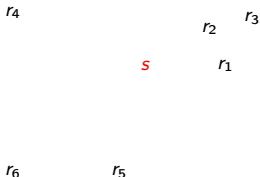


→ awake robots can move and wake up sleeping robots

The Freeze-Tag Problem [Arkin et al. SODA '02]

Input:

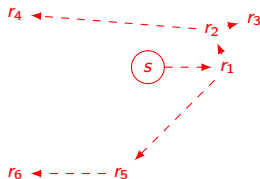
- n sleeping robots in the plane
- robot s , initially awake



→ awake robots can move and wake up sleeping robots

Output:

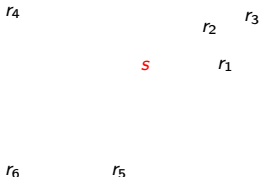
- A wake-up tree t rooted at s



The Freeze-Tag Problem [Arkin et al. SODA '02]

Input:

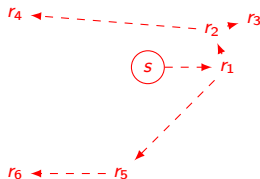
- n sleeping robots in the plane
- robot s , initially awake



→ awake robots can move and wake up sleeping robots

Output:

- A wake-up tree t rooted at s



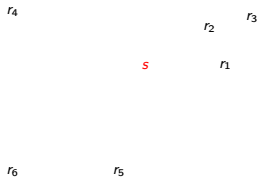
→ Optimization problem:

Performance \equiv weighted depth of t
 \equiv **makespan** (as small as possible)

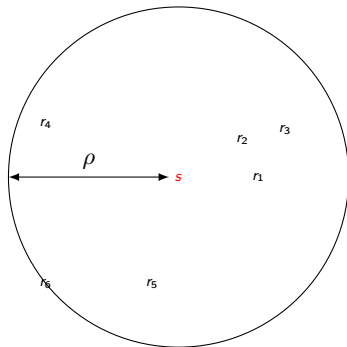
Bounds for the centralized FTP

[Abel, Akitiya, Yu, FWCG '17]

Finding optimal wake-up tree t^* is NP-hard.



Bounds for the centralized FTP

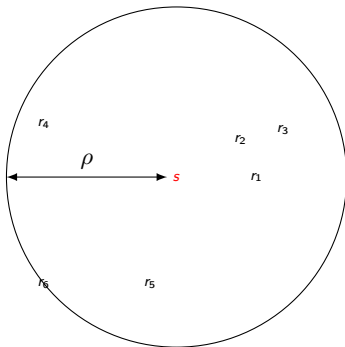


[Abel, Akitiya, Yu, FWCG '17]

Finding optimal wake-up tree t^* is NP-hard.

Remark: depth of t^* is at least ρ .

Bounds for the centralized FTP



[Abel, Akitiya, Yu, FWCG '17]

Finding optimal wake-up tree t^* is NP-hard.

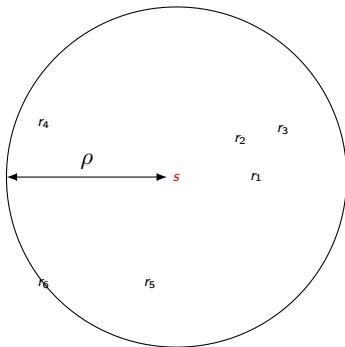
Remark: depth of t^* is at least ρ .

[Bonichon, Gavoille, Hanusse, Odak, CCCG '24]

Wake-up tree with depth at most 4.63ρ can be computed in time $O(n)$.

\Rightarrow linear approximation of the optimal: ✓

Bounds for the centralized FTP



[Abel, Akitiya, Yu, FWCG '17]

Finding optimal wake-up tree t^* is NP-hard.

Remark: depth of t^* is at least ρ .

[Bonichon, Gavoille, Hanusse, Odak, CCCG '24]

Wake-up tree with depth at most 4.63ρ can be computed in time $O(n)$.

\Rightarrow linear approximation of the optimal: ✓

Conjecture [B. Casteigts, G. H. DISC '24]

Worst case: 4 robots, positioned in a square.

Depth of t^* is $(1 + 2\sqrt{2})\rho \simeq 3.83\rho$.

Proven for $n \notin [8, 280]$, and for convex positioned robots.

Distributed Freeze-Tag

Distributed Setting

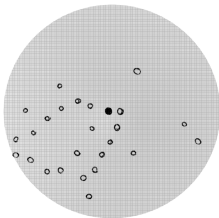
Framework: no global knowledge & no global computation.

Distributed Freeze-Tag

Distributed Setting

Framework: no global knowledge & no global computation.

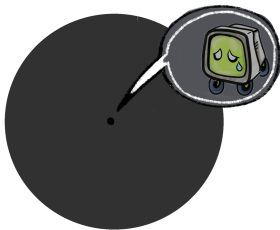
- n initially sleeping robots, source s



Distributed Freeze-Tag

Distributed Setting

Framework: no global knowledge & no global computation.

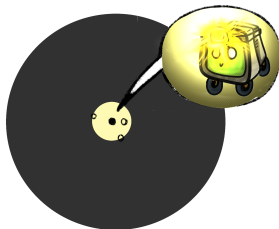


- n initially sleeping robots, source s
- No central knowledge: positions of sleeping robots are unknown

Distributed Freeze-Tag

Distributed Setting

Framework: no global knowledge & no global computation.

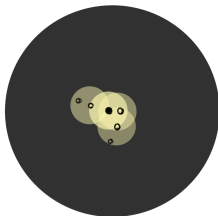


- n initially sleeping robots, source s
- No central knowledge: positions of sleeping robots are unknown
- Detection (snapshot) with constant radius
- No distant communication

Distributed Freeze-Tag

Distributed Setting

Framework: no global knowledge & no global computation.

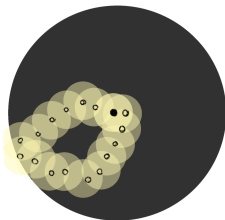


- n initially sleeping robots, source s
- No central knowledge: positions of sleeping robots are unknown
- Detection (snapshot) with constant radius
- No distant communication
- Robots can progressively wake-up the entire swarm

Distributed Freeze-Tag

Distributed Setting

Framework: no global knowledge & no global computation.

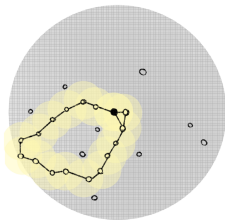


- n initially sleeping robots, source s
- No central knowledge: positions of sleeping robots are unknown
- Detection (snapshot) with constant radius
- No distant communication
- Robots can progressively wake-up the entire swarm

Distributed Freeze-Tag

Distributed Setting

Framework: no global knowledge & no global computation.

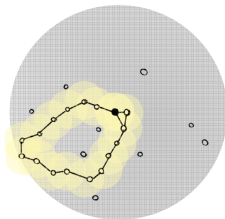


- n initially sleeping robots, source s
- No central knowledge: positions of sleeping robots are unknown
- Detection (snapshot) with constant radius
- No distant communication
- Robots can progressively wake-up the entire swarm **or can they ?**

Distributed Freeze-Tag

Distributed Setting

Framework: no global knowledge & no global computation.



- n initially sleeping robots, source s
- No central knowledge: positions of sleeping robots are unknown
- Detection (snapshot) with constant radius
- No distant communication
- Robots can progressively wake-up the entire swarm **or can they ?**
- Robots may need to explore further than the unit-disk graph.

Notations

ℓ -disk graph

$$\{r_1, r_2\} \in E \iff |r_1 r_2| \leq \ell$$

Notations

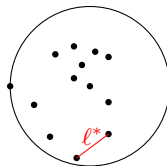
ℓ -disk graph

$$\{r_1, r_2\} \in E \iff |r_1 r_2| \leq \ell$$

Connectivity threshold

Connectivity threshold $\ell^* \equiv$
smallest $\ell > 0$ such that the ℓ -disk
graph is connected.

Hypothesis: the algorithm is given
an upper bound ℓ on ℓ^* .



Notations

 ℓ -disk graph

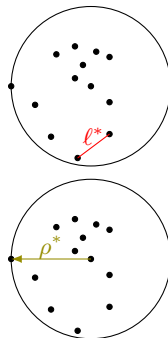
$$\{r_1, r_2\} \in E \iff |r_1 r_2| \leq \ell$$

Connectivity threshold

Connectivity threshold $\ell^* \equiv$
smallest $\ell > 0$ such that the ℓ -disk
graph is connected.

Hypothesis: the algorithm is given
an upper bound ℓ on ℓ^* .

Bonus Hypothesis: the algorithm is
given an upper bound ρ on the
radius ρ^* .



Notations

 ℓ -disk graph

$$\{r_1, r_2\} \in E \iff |r_1 r_2| \leq \ell$$

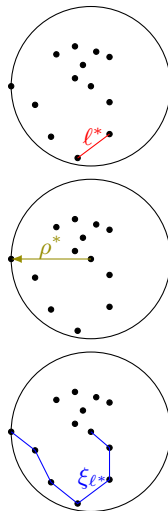
Connectivity threshold

Connectivity threshold $\ell^* \equiv$ smallest $\ell > 0$ such that the ℓ -disk graph is connected.

Hypothesis: the algorithm is given an upper bound ℓ on ℓ^* .

Bonus Hypothesis: the algorithm is given an upper bound ρ on the radius ρ^* .

Additional notation: ξ_ℓ is the eccentricity of the ℓ -disk graph.



Contribution

Energy	Makespan	Lower Bound
unconstrained	$O(\rho + \ell^2 \log(\rho/\ell))$	$\Omega(\rho + \ell^2 \log(\rho/\ell))$
$< \pi(\ell^2 - 1)/2$	-	unfeasible
$\Theta(\ell^2)$	$O(\xi_\ell \cdot \ell)$	$\Omega(\xi_\ell + \ell^2 \log(\xi_\ell/\ell))$
$\Theta(\ell^2 \log \ell)$	$O(\xi_\ell + \ell^2 \log(\xi_\ell/\ell))$	

Table: Complexity of the makespan for the dFTP given (ℓ, ρ, n) .

$\rho \equiv$ radius $\ell \equiv$ connectivity threshold $\xi_\ell \equiv$ eccentricity of the ℓ -disk graph

Contribution

Energy	Makespan	Lower Bound
unconstrained	$O(\rho + \ell^2 \log(\rho/\ell))$	$\Omega(\rho + \ell^2 \log(\rho/\ell))$
$< \pi(\ell^2 - 1)/2$	-	unfeasible
$\Theta(\ell^2)$	$O(\xi_\ell \cdot \ell)$	$\Omega(\xi_\ell + \ell^2 \log(\xi_\ell/\ell))$
$\Theta(\ell^2 \log \ell)$	$O(\xi_\ell + \ell^2 \log(\xi_\ell/\ell))$	

Table: Complexity of the makespan for the dFTP given (ℓ, ρ, n) .

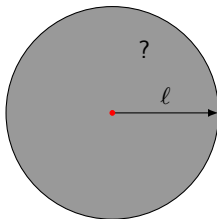
$\rho \equiv$ radius $\ell \equiv$ connectivity threshold $\xi_\ell \equiv$ eccentricity of the ℓ -disk graph

Comparison with the centralized setting

- linear in ρ
- requires additional term $O(\ell^2 \log \rho/\ell)$

Recruitment

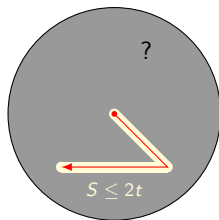
Goal: Wake-up new robots



Recruitment

Goal: Wake-up new robots
Find

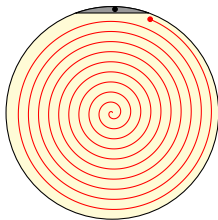
- Finding one robot in the ℓ -disk requires exploring



Recruitment

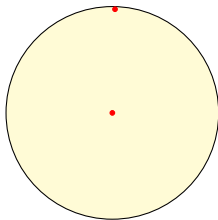
Goal: Wake-up new robots
Find

- Finding one robot in the ℓ -disk requires exploring a surface of $O(\ell^2)$.



Recruitment

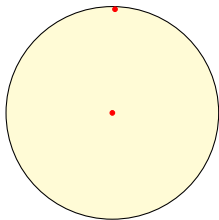
Goal: Wake-up new robots
Find



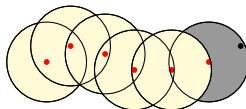
- Finding one robot in the ℓ -disk requires exploring a surface of $O(\ell^2)$.
- Takes time $O(\ell^2)$ to a single robot... *and then they were 2.*

Recruitment

Goal: Wake-up new robots
Find

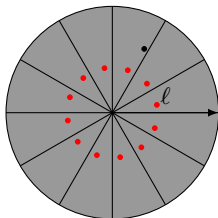
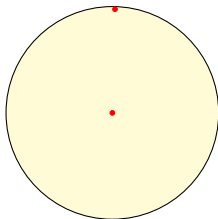


- Finding one robot in the ℓ -disk requires exploring a surface of $O(\ell^2)$.
- Takes time $O(\ell^2)$ to a single robot... *and then they were 2.*
- Can be iterated (DFS)

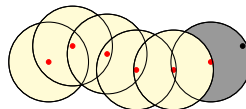


Recruitment

Goal: Wake-up new robots
Find



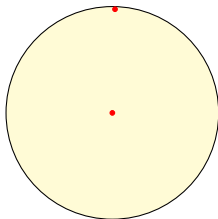
- Finding one robot in the ℓ -disk requires exploring a surface of $O(\ell^2)$.
- Takes time $O(\ell^2)$ to a single robot... *and then they were 2.*
- Can be iterated (DFS)



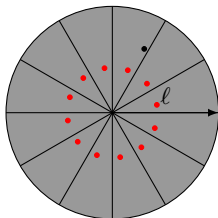
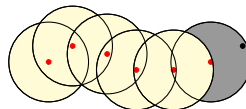
- A team of k robots can explore such surface in time $O(\ell + \ell^2/k)$.

Recruitment

Goal: Wake-up new robots
Find



- Finding one robot in the ℓ -disk requires exploring a surface of $O(\ell^2)$.
- Takes time $O(\ell^2)$ to a single robot... *and then they were 2.*
- Can be iterated (DFS)



- A team of k robots can explore such surface in time $O(\ell + \ell^2/k)$.

First Team

\Rightarrow it takes time $\sum_{k=1}^{4\ell} (\ell + \ell^2/k) = O(\ell^2 \log \ell)$ to wake up a team of size 4ℓ starting with 1 robot.

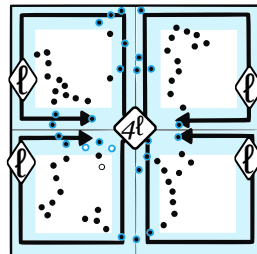
Geometric Separators

Starting with a team of size 4ℓ

Divide and Conquer

(Recursively) divide the surface assigned to a given team, using *geometric separators* of width ℓ .

- Geometric separator \sim cut of the ℓ -disk graph



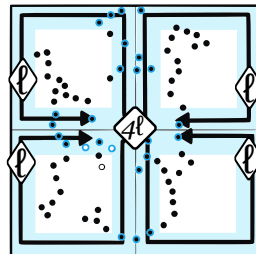
Geometric Separators

Starting with a team of size 4ℓ

Divide and Conquer

(Recursively) divide the surface assigned to a given team, using *geometric separators* of width ℓ .

- Geometric separator \sim cut of the ℓ -disk graph
- Detects if a large area is empty (since $\ell \geq \ell^*$)



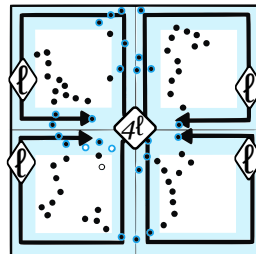
Geometric Separators

Starting with a team of size 4ℓ

Divide and Conquer

(Recursively) divide the surface assigned to a given team, using *geometric separators* of width ℓ .

- Geometric separator \sim cut of the ℓ -disk graph
- Detects if a large area is empty (since $\ell \geq \ell^*$)
- Breaks the ℓ -disk graph $\Rightarrow o(\xi)$ possible



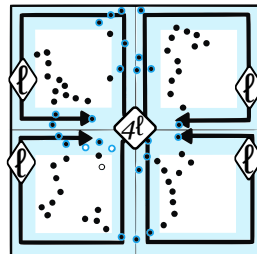
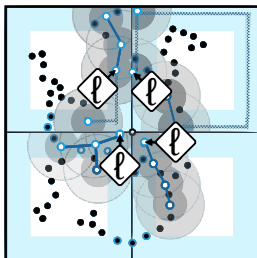
Geometric Separators

Starting with a team of size 4ℓ

Divide and Conquer

(Recursively) divide the surface assigned to a given team, using *geometric separators* of width ℓ .

- Geometric separator \sim cut of the ℓ -disk graph
- Detects if a large area is empty (since $\ell \geq \ell^*$)
- Breaks the ℓ -disk graph $\Rightarrow o(\xi)$ possible



Rebuild teams: $\ell \rightarrow 4\ell$

- Recruiting 3ℓ additional robots in its square can lead to a "big" increase of the distance to the sleeping robots round after round.

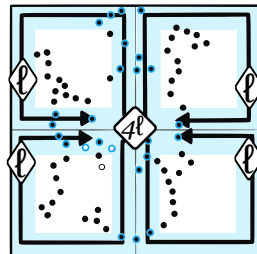
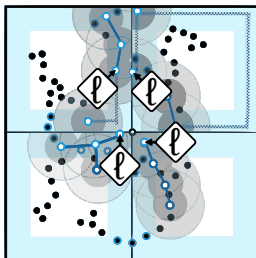
Geometric Separators

Starting with a team of size 4ℓ

Divide and Conquer

(Recursively) divide the surface assigned to a given team, using *geometric separators* of width ℓ .

- Geometric separator \sim cut of the ℓ -disk graph
- Detects if a large area is empty (since $\ell \geq \ell^*$)
- Breaks the ℓ -disk graph $\Rightarrow o(\xi)$ possible

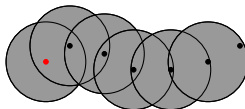


Rebuild teams: $\ell \rightarrow 4\ell$

- Recruiting 3ℓ additional robots in its square can lead to a "big" increase of the distance to the sleeping robots round after round.
- Instead, each team makes sure that 4ℓ robots *have been awoken* in its region.

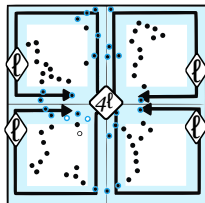
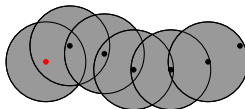
Algorithm

1 - Recruit a team of 4ℓ robots and position it in the center of the region



Algorithm

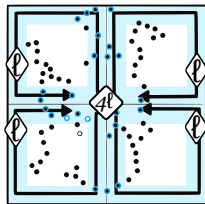
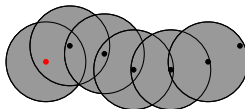
1 - Recruit a team of 4ℓ robots and position it in the center of the region



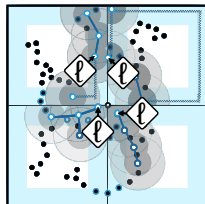
2 - Constitute 4 teams of size ℓ and explore the separator of the sub-square.

Algorithm

1 - Recruit a team of 4ℓ robots and position it in the center of the region



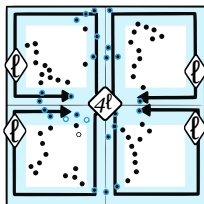
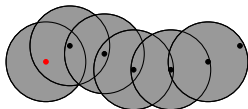
2 - Constitute 4 teams of size ℓ and explore the separator of the sub-square.



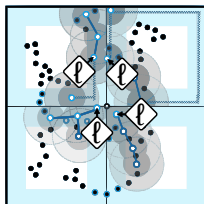
3 - Each team recruits new robots until 4ℓ robots initially positioned in the sub-square are awoken.

Algorithm

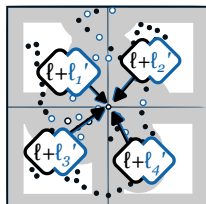
1 - Recruit a team of 4ℓ robots and position it in the center of the region



2 - Constitute 4 teams of size ℓ and explore the separator of the sub-square.



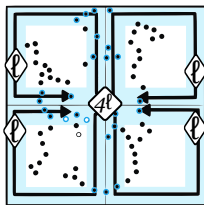
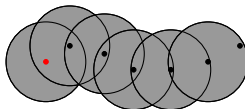
3 - Each team recruits new robots until 4ℓ robots initially positioned in the sub-square are awoken.



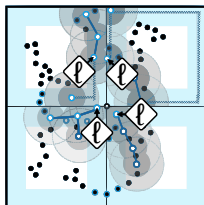
4 - All 4 teams meet in the center of the original region.

Algorithm

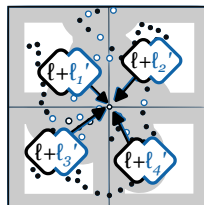
1 - Recruit a team of 4ℓ robots and position it in the center of the region



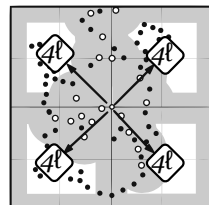
2 - Constitute 4 teams of size ℓ and explore the separator of the sub-square.



3 - Each team recruits new robots until 4ℓ robots initially positioned in the sub-square are awoken.



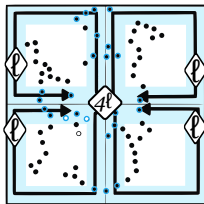
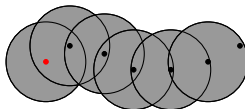
4 - All 4 teams meet in the center of the original region.



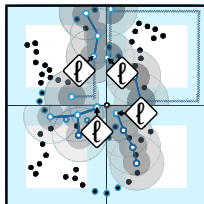
5 - Every robot goes to its origin region

Algorithm

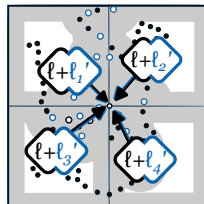
1 - Recruit a team of 4ℓ robots and position it in the center of the region



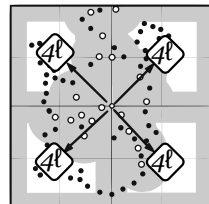
2 - Constitute 4 teams of size ℓ and explore the separator of the sub-square.



3 - Each team recruits new robots until 4ℓ robots initially positioned in the sub-square are awoken.



4 - All 4 teams meet in the center of the original region.



5 - Every robot goes to its origin region and GOTO 2.

Details on the Recruitment

Non-optimality

First recruitment $\in O(\ell^2 \log \ell)$

Details on the Recruitment

Non-optimality

First recruitment $\in O(\ell^2 \log \ell) \in \omega(\ell^2 \log \rho / \ell)$ if $\ell \sim \rho$

Details on the Recruitment

Non-optimality

First recruitment $\in O(\ell^2 \log \ell) \in \omega(\ell^2 \log \rho / \ell)$ if $\ell \sim \rho$

Depth-First Sampling

To achieve optimality: take advantage of the recruitment process to explore an area as large as possible, with early termination whenever $\ell^3 > \rho^2$.

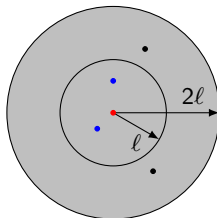
Details on the Recruitment

Non-optimality

First recruitment $\in O(\ell^2 \log \ell) \in \omega(\ell^2 \log \rho / \ell)$ if $\ell \sim \rho$

Depth-First Sampling

To achieve optimality: take advantage of the recruitment process to explore an area as large as possible, with early termination whenever $\ell^3 > \rho^2$.



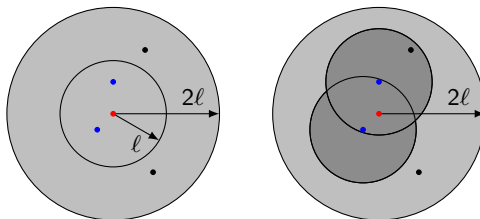
Details on the Recruitment

Non-optimality

First recruitment $\in O(\ell^2 \log \ell) \in \omega(\ell^2 \log \rho / \ell)$ if $\ell \sim \rho$

Depth-First Sampling

To achieve optimality: take advantage of the recruitment process to explore an area as large as possible, with early termination whenever $\ell^3 > \rho^2$.



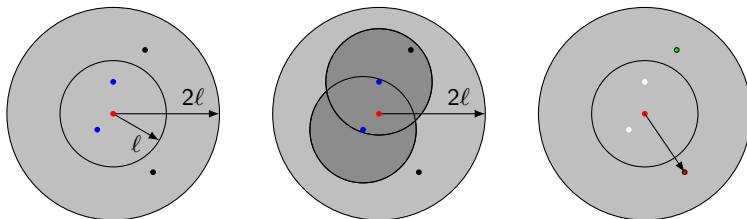
Details on the Recruitment

Non-optimality

First recruitment $\in O(\ell^2 \log \ell) \in \omega(\ell^2 \log \rho / \ell)$ if $\ell \sim \rho$

Depth-First Sampling

To achieve optimality: take advantage of the recruitment process to explore an area as large as possible, with early termination whenever $\ell^3 > \rho^2$.



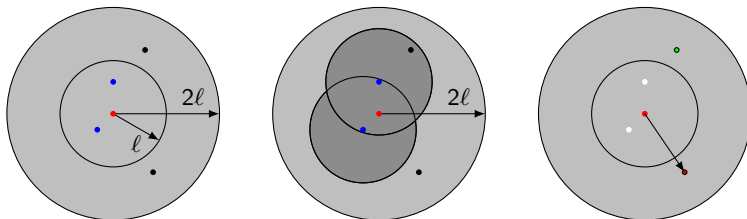
Details on the Recruitment

Non-optimality

First recruitment $\in O(\ell^2 \log \ell) \in \omega(\ell^2 \log \rho / \ell)$ if $\ell \sim \rho$

Depth-First Sampling

To achieve optimality: take advantage of the recruitment process to explore an area as large as possible, with early termination whenever $\ell^3 > \rho^2$.



- Two recruited robots are at distance at least ℓ .
- There exists at most $O(\rho^2 / \ell^2)$ such robots

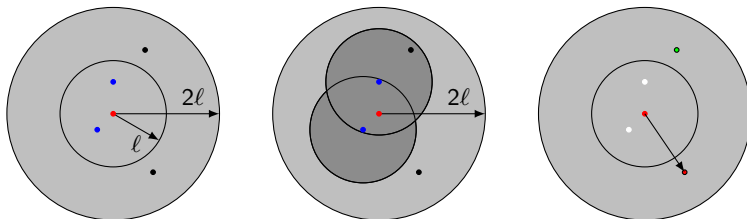
Details on the Recruitment

Non-optimality

First recruitment $\in O(\ell^2 \log \ell) \in \omega(\ell^2 \log \rho / \ell)$ if $\ell \sim \rho$

Depth-First Sampling

To achieve optimality: take advantage of the recruitment process to explore an area as large as possible, with early termination whenever $\ell^3 > \rho^2$.



- Two recruited robots are at distance at least ℓ .
- There exists at most $O(\rho^2 / \ell^2)$ such robots

Theorem

The execution of Phase 1 with this procedure takes time $O(\rho + \ell^2 \log \rho / \ell)$

Conclusion

Results

Energy	Makespan	Lower Bound
unconstrained	$O(\rho + \ell^2 \log(\rho/\ell))$	$\Omega(\rho + \ell^2 \log(\rho/\ell))$
$< \pi(\ell^2 - 1)/2$	-	unfeasible
$\Theta(\ell^2)$	$O(\xi_\ell \cdot \ell)$	$\Omega(\xi_\ell + \ell^2 \log(\xi_\ell/\ell))$
$\Theta(\ell^2 \log \ell)$	$O(\xi_\ell + \ell^2 \log(\xi_\ell/\ell))$	

Conclusion

Results

Energy	Makespan	Lower Bound
unconstrained	$O(\rho + \ell^2 \log(\rho/\ell))$	$\Omega(\rho + \ell^2 \log(\rho/\ell))$
$< \pi(\ell^2 - 1)/2$	-	unfeasible
$\Theta(\ell^2)$	$O(\xi_\ell \cdot \ell)$	$\Omega(\xi_\ell + \ell^2 \log(\xi_\ell/\ell))$
$\Theta(\ell^2 \log \ell)$	$O(\xi_\ell + \ell^2 \log(\xi_\ell/\ell))$	

Further works

- Without the knowledge of ℓ
- Efficient algorithm with $\Theta(\ell^2)$ energy, or more precise lower bound
- Several sources with different spacial and/or temporal reference
- Under asynchronous assumptions
- Minimization of the global energy

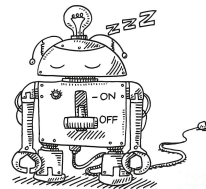
Conclusion

Results

Energy	Makespan	Lower Bound
unconstrained	$O(\rho + \ell^2 \log(\rho/\ell))$	$\Omega(\rho + \ell^2 \log(\rho/\ell))$
$< \pi(\ell^2 - 1)/2$	-	unfeasible
$\Theta(\ell^2)$	$O(\xi_\ell \cdot \ell)$	$\Omega(\xi_\ell + \ell^2 \log(\xi_\ell/\ell))$
$\Theta(\ell^2 \log \ell)$	$O(\xi_\ell + \ell^2 \log(\xi_\ell/\ell))$	

Further works

- Without the knowledge of ℓ
- Efficient algorithm with $\Theta(\ell^2)$ energy, or more precise lower bound
- Several sources with different spacial and/or temporal reference
- Under asynchronous assumptions
- Minimization of the global energy



Thanks!