Collaborative Exploration by Energy-Constrained Mobile Robots

Shantanu Das[1] Dariusz Dereniowski[2] Christina Karousatou[1]

[1]LIF, Aix-Marseille University and CNRS, Marseille, France

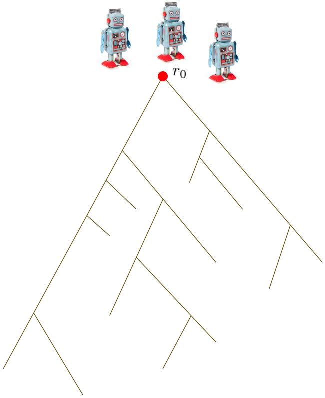
[2]Faculty of Electronics, Telecommunications and Informatics, Gdańsk University of Technology, Gdańsk, Poland

JGA 2015, Orléans, France

Graph Exploration

- We have multiple energy constrained mobile agents.
- Each node has to be visited by at least one agent.

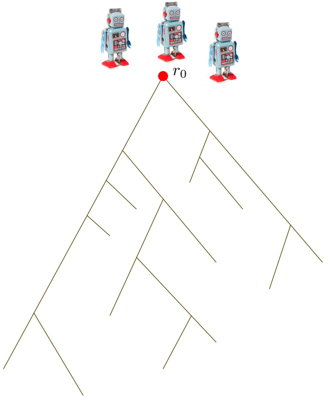
Goal: Minimize the number of agents used.



Graph Exploration

- We have multiple energy constrained mobile agents.
- Each node has to be visited by at least one agent.

Goal: Minimize the number of agents used.



Motivation

Real robots have limited energy!

The model

- Anonymous rooted tree T with local port numbering.
- Tree's nodes are memoryless.
- Root r_0 contains an infinite supply of mobile agents.
    - But we will only use a few!
- Each mobile agent has:
    - Limited energy B, 1 edge traversal = 1 energy unit.
    - Unique identity.
    - Unlimited memory.

We consider two communication scenarios:

- Global communication model.
  - An agent can communicate with any other agent instantaneously.
- Local communication model.
  - Any two agents must be simultaneously on the same node to exchange information.

We consider two communication scenarios:

- Global communication model.
  - An agent can communicate with any other agent instantaneously.
- Local communication model.
  - Any two agents must be simultaneously on the same node to exchange information.

Remark: Tree's height has to be at most B!

Piecemeal Exploration

- For a single energy constrained agent, where refuelling is allowed, exploration can be performed in $O(|E|)$ steps [*C. Duncan et al.* '06].

Piecemeal Exploration

- For a single energy constrained agent, where refuelling is allowed, exploration can be performed in $O(|E|)$ steps [*C. Duncan et al.* '06].

Offline Multiple Agent Exploration

- Exploration with k agents each having B units of available energy.
- Optimizing either k or B is NP-hard. [*P. Fraignaud et al.* '06].

Online Exploration

- The *competitive ratio* is used to measure the efficiency of online algorithms.
    - It is equal to the worst case ratio of the cost of an online algorithm for some graph G over the cost of the optimal offline algorithm for the same graph.

- For a fixed number of agents k, the goal is to minimize the energy needed to perform the exploration. [*Dynia M. et al.* '07]
    - Proved a lower bound of 3/2.
    - Provided a $4 - 2/k$ competitive algorithm.

Our Contributions

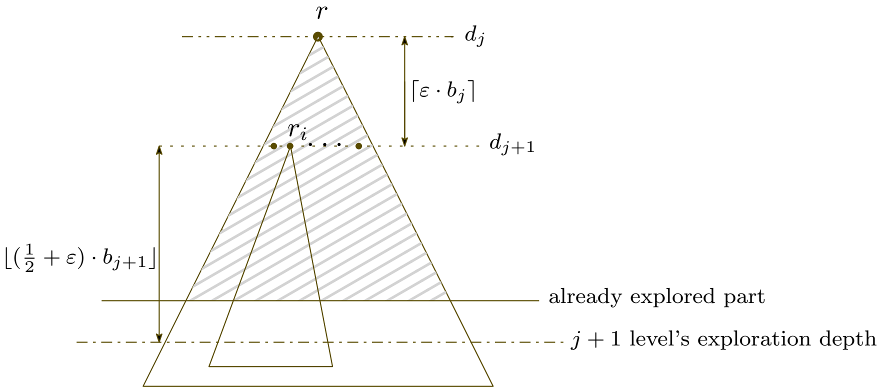
- We provide an online algorithm under the global communication scenario with a competitive ratio of $O(\log B)$.
- We modify the online algorithm to work under the local communication scenario and prove that the competitive ratio is preserved.
- We prove an $\Omega(\log B)$ lower bound on the competitive ratio for any online algorithm for the local communication model.

Main ideas of the algorithm:

- Explore the tree by levels.
- At each level expand the exploration up until a certain depth.
- Recursively call the algorithm for each node at the next level.

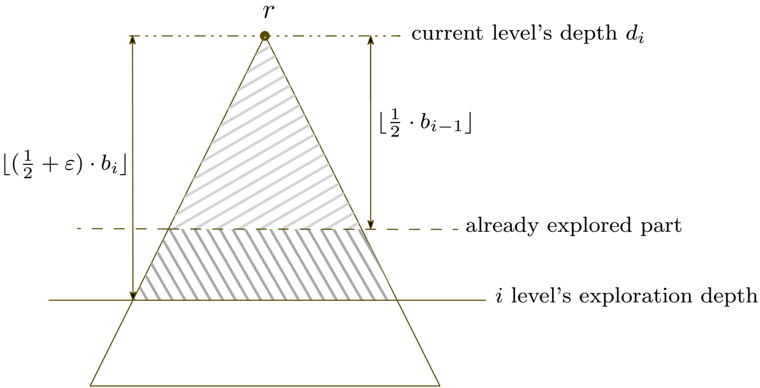
$GCTE_\varepsilon(r, b)$ Algorithm, $0 < \varepsilon < \frac{1}{4}$

- Uncover $(r, \lfloor(\frac{1}{2} + \varepsilon)b\rfloor)$.
- Let $r_1, r_2, \ldots$ be nodes at depth $\lceil \varepsilon \cdot b \rceil$ from r, such that T_{r_i} has some unexplored edges.
- For each r_i call $GCTE_\varepsilon(r_i, (b - \lceil \varepsilon \cdot b \rceil))$.

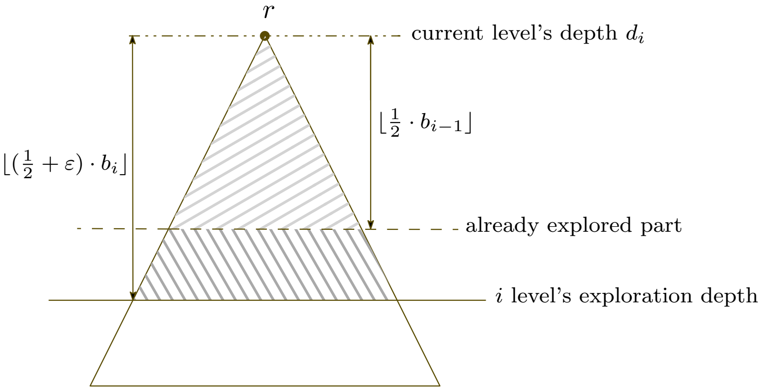


$Uncover(r, \delta)$ Procedure

- The agents explore the subtree using a DFS traversal.
- The exploration is restricted to a depth of δ from r.
- When an agent has $x(\varepsilon) = \frac{1}{2}(\frac{1}{2} - \varepsilon)b$ units of energy left stops the exploration.
- The next agent continues the exploration according to the DFS traversal.
- The procedure ends when all the nodes at depth δ or less have been visited.



r
current level's depth d_i
$\lfloor \frac{1}{2} \cdot b_{i-1} \rfloor$
$\lfloor (\frac{1}{2} + \varepsilon) \cdot b_i \rfloor$
already explored part
i level's exploration depth



Note

Any agent that reaches the unexplored part of a subtree T_r cannot return to node r.

Properties of $GCTE_\varepsilon(r,b)$:

- At each level the algorithm expands the explored part $(\lceil \varepsilon \cdot b_i \rceil)$.
- The number of levels in $GCTE_\varepsilon(r,b)$ is at most $\log_{\frac{1}{1-\varepsilon}} B$.
- During each call of procedure $Uncover(r,\delta)$ at most $\frac{4}{(\frac{1}{2}-\varepsilon)} \cdot OPT$ agents are used.
- $GCTE_\varepsilon$ has a competitive ratio of $\frac{4}{(\frac{1}{2}-\varepsilon)} \cdot \log_{(\frac{1}{1-\varepsilon})} B$.

Adapting $GCTE_{\varepsilon}$ to the local communication model.

There are two issues to overcome:

- How do the agents meet to exchange information?
  - During the *Uncover* procedure.
  - Between the levels of the algorithm.
- How do the agents know when to start from the global root?

How to meet ?

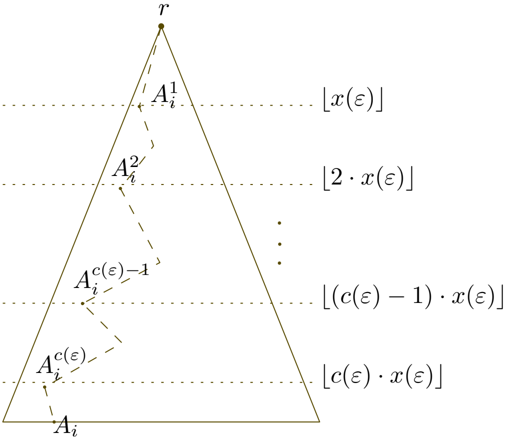
Suppose that k agents $(A_1, \ldots, A_k)$ are needed to perform the *Uncover* procedure during a call of $GCTE_\varepsilon(r, b)$ with global communication.

How to meet ?

Suppose that k agents $(A_1, \ldots, A_k)$ are needed to perform the $Uncover$ procedure during a call of $GCTE_\varepsilon(r, b)$ with global communication.

- For each agent A_i needed, we add a constant number of $c(\varepsilon) = 2 \cdot \left\lceil \frac{1/2+\varepsilon}{1/2-\varepsilon} \right\rceil$ extra agents, called the i-th team. $(A_i^1, \ldots, A_i^{c(\varepsilon)})$
- A team begins the exploration only if all the $c(\varepsilon) + 1$ agents are located in the root and if the previous team has finished its movements.

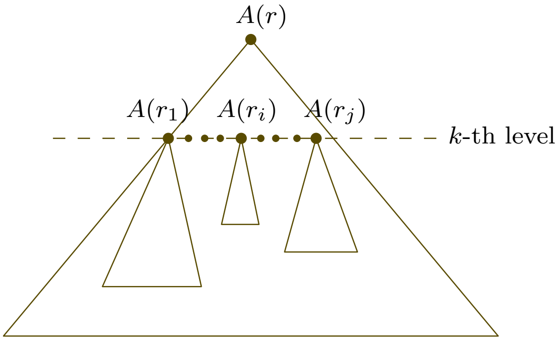
- Each extra agent A_i^j follows A_i's movements until depth $d_j(\varepsilon) = \lfloor j \cdot x(\varepsilon) \rfloor$.
- When A_i interrupts its exploration, it heads towards the root.
- When A_i meets $A_i^{c(\varepsilon)}$, $A_i^{c(\varepsilon)}$ begins to head towards the root too.
- Whenever an agent A_i^j meets its ancestor A_i^{j-1}, they both head towards the root.
- Eventually, A_i^1 reaches the root and the next team can resume the exploration.


r
A_i^1
$\lfloor x(\varepsilon) \rfloor$
A_i^2
$\lfloor 2 \cdot x(\varepsilon) \rfloor$
$A_i^{c(\varepsilon)-1}$
$\lfloor (c(\varepsilon)-1) \cdot x(\varepsilon) \rfloor$
$A_i^{c(\varepsilon)}$
$\lfloor c(\varepsilon) \cdot x(\varepsilon) \rfloor$
A_i

Communication between levels is performed by the *managing agents*.

Every subtree T_r has a managing agent $A(r)$ located at the root r. Its tasks are:

- Redirect the agents to the correct subtree.
- Once the exploration of T_r is finished, report it to its ancestor.


$A(r)$
$A(r_1)$
$A(r_i)$
$A(r_j)$
k-th level

When to start ?

- $A(r_0)$ has all the agents at its disposal.
- To avoid using more agents than needed we introduce an artificial delay $d(\varepsilon) = (c(\varepsilon) + 2)B$.
- $d(\varepsilon)$ denotes the time intervals during which a new agent will start from the global root r_0.

Counting the number of additional agents used.

- For each agent used in $GCTE_\varepsilon$ during the Uncover procedure, we use a constant number of extra agents $c(\varepsilon)$, to perform the extended DFS traversal.
- The overall number of managing agents is bounded by $\log_{(\frac{1}{1-\varepsilon})} B \cdot OPT$

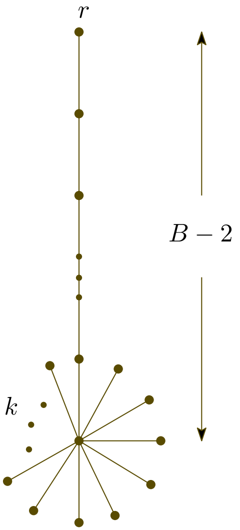
Counting the number of additional agents used.

- For each agent used in $GCTE_\varepsilon$ during the Uncover procedure, we use a constant number of extra agents $c(\varepsilon)$, to perform the extended DFS traversal.
- The overall number of managing agents is bounded by $\log_{(\frac{1}{1-\varepsilon})} B \cdot OPT$

Hence, the adaptation of $GCTE_\varepsilon$ under the local communication scenario preserves the competitive ratio of $O(\log B)$.

A bad family of instances:

- An offline algorithm will use exactly k agents.
- In any online algorithm at least $\Omega(\log B)$ agents are needed only for carrying the information from the node at depth $B-2$ to the root.
- For $k=1$ we get an $\Omega(\log B)$ competitive ratio for any online algorithm.



- Investigate whether more efficient algorithms are possible for the global communication scenario.
- Study the case of general graphs and other topologies.

- Investigate whether more efficient algorithms are possible for the global communication scenario.
- Study the case of general graphs and other topologies.

...Merci!