How to beat a random walk with a clock ?

Locating a Target With an Agent Guided by Unreliable Local Advice

Nicolas Hanusse[1] David Ilcinkas[1] Adrian Kosowski[1,3]
Nicolas Nisse[2]

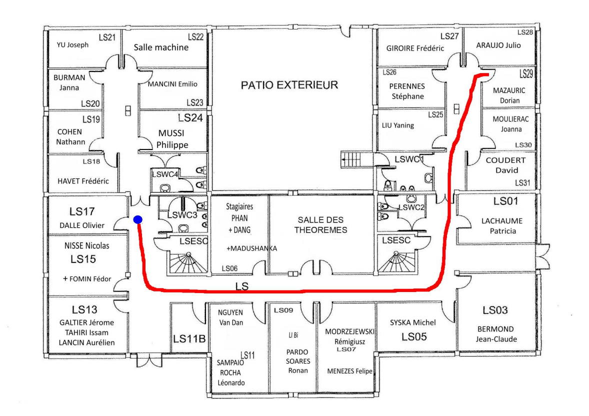
[1]CNRS-LaBRI, Université Bordeaux, Projet INRIA Cépage
[2]INRIA Sophia-Antipolis, Projet Mascotte
[3]Gdańsk University of Technology, Poland

Séminar MASCOTTE

Sophia Antipolis, March 15th, 2011

Many thanks to Nicolas Hanusse for his slides

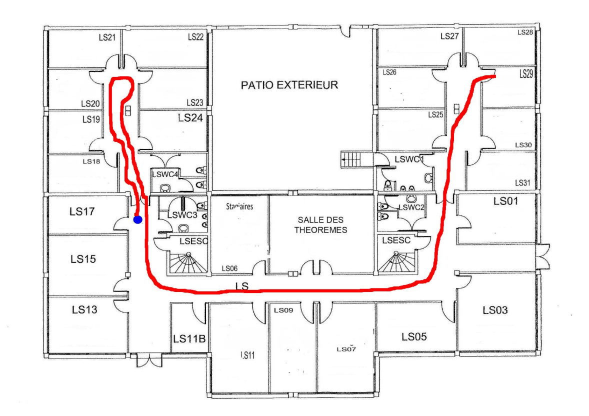
How to find Dorian in MASCOTTE's floor?



Take advantage of your knowledge

Map, sense of direction (left/right, north/south...), topology...

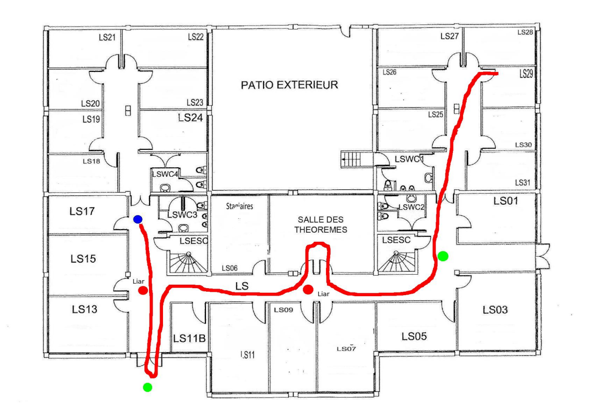
How to find Dorian in MASCOTTE's floor?



Take advantage of your knowledge

Map, sense of direction (left/right, north/south...), topology...

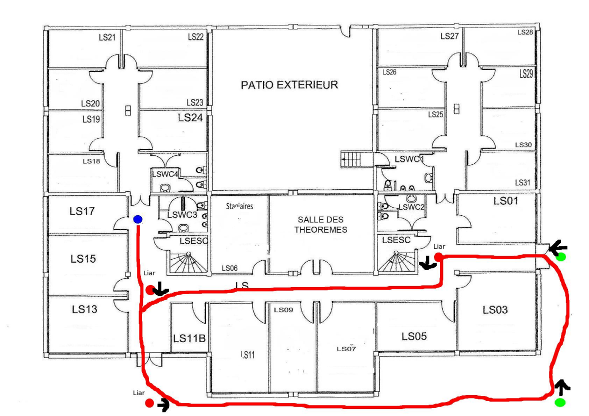
How to find Dorian in MASCOTTE's floor?



Algorithm A (Advice)

Keep on following the advice until the target t is found.

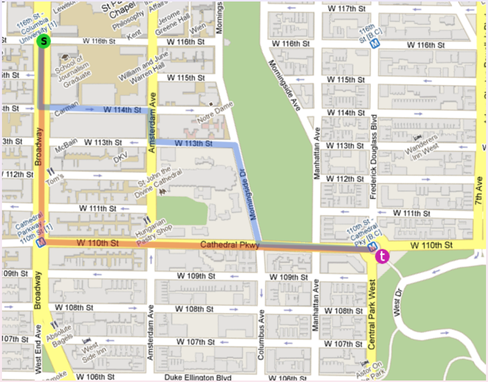
How to find Dorian in MASCOTTE's floor?



Algorithm A (Advice)

Keep on following the advice until the target t is found (LOOP).

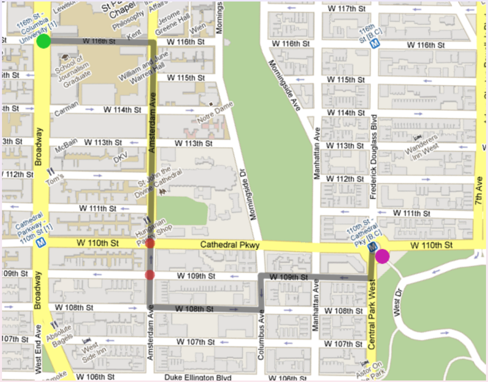
How to find a cash machine in NY ?



Take advantage of your knowledge

Map, sense of direction (left/right, north/south...), topology...

How to find a cash machine in NY ?



Algorithm A (Advice)

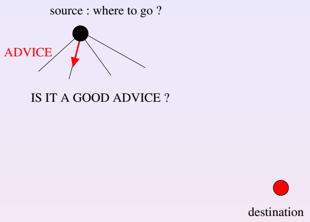
Keep on following the advice until the target t is found.

How to find a cash machine in NY ?

Algorithm A (Advice)

Keep on following the advice until the target t is found (LOOP).

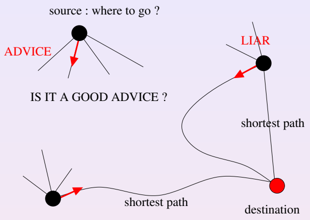
Searching for information in networks with liars



Numerous adversaries

- the network map and the target location *unknown*
- dynamicity $\Longrightarrow$ local information unreliable

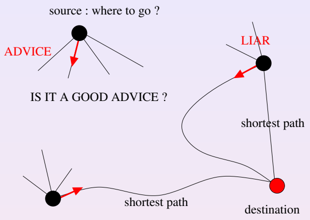
Searching for information in networks with liars



Numerous adversaries

- the network map and the target location *unknown*
- dynamicity $\Longrightarrow$ local information unreliable

Searching for information in networks with liars



Numerous adversaries

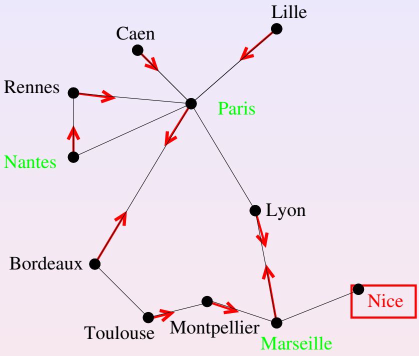
- the network map and the target location *unknown*
- dynamicity $\Longrightarrow$ local information unreliable

Some models of searching with errors

For every target t, each node gives an advice. If this advice is bad, the node is a liar.

- *Searching with uncertainty*, SIROCCO'1999, Kranakis-Krizanc
- *Searching with mobile agents in networks with liars*, Disc. Applied Maths. 2004, Hanusse-Kranakis-Krizanc

A model of searching with errors



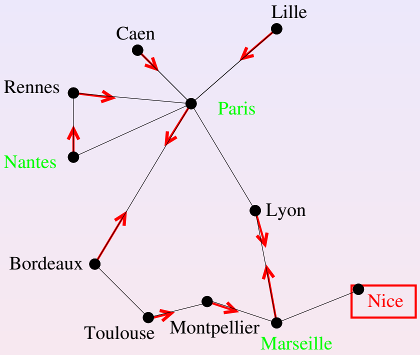
Local information

- **advice**: For each target t, every node u points to an incident link e;
- If e is on a shortest path from u to t, u is a truthteller , otherwise a liar.

Hypothesis

- Advice and topology are *unchanged* during the search;
- **worst-case analysis**: The *adversary* knows the algorithm and chooses the worst configuration of advice.

A model of searching with errors



Local information

- **advice**: For each target t, every node u points to an incident link e;
- If e is on a shortest path from u to t, u is a truthteller , otherwise a liar.

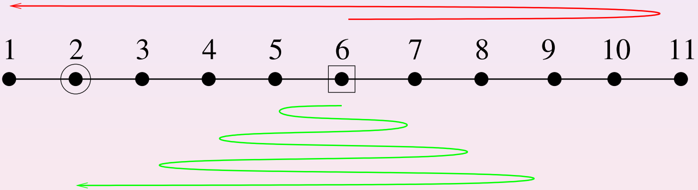
Hypothesis

- Advice and topology are *unchanged* during the search;
- **worst-case analysis**: The *adversary* knows the algorithm and chooses the worst configuration of advice.

Performance Measures

Performance Measures

- n: # nodes, k: # liars, d: distance to the target
- **Time** $\mathcal{T}$: # hops to reach the target
- **Memory** $\mathcal{M}$: of the mobile agent



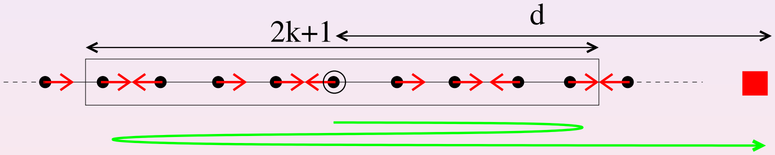
With sense of direction:

- $\mathcal{T} \leq 2n$ et $\mathcal{M} = \Theta(1)$ - whole exploration
- $\mathcal{T} = \Theta(d)$ et $\mathcal{M} = \Theta(\log n)$ - zigzag
- $\mathcal{T} \leq d + 4k + 2$ et $\mathcal{M} = \Theta(\log k)$ - using advice, know k

Performance Measures

Performance Measures

- n: # nodes, k: # liars, d: distance to the target
- **Time** $\mathcal{T}$: # hops to reach the target
- **Memory** $\mathcal{M}$: of the mobile agent



With sense of direction:

- $\mathcal{T} \leq 2n$ et $\mathcal{M} = \Theta(1)$ - whole exploration
- $\mathcal{T} = \Theta(d)$ et $\mathcal{M} = \Theta(\log n)$ - zigzag
- $\mathcal{T} \leq d + 4k + 2$ et $\mathcal{M} = \Theta(\log k)$ - using advice, know k

State of the art

2 types of results

- *specialized deterministic* algorithms: ring, hypercube, complete graph, ...
- *universal randomized* algorithms with $\mathcal{M} = 0$: random walk (R), biasest random walk (BR)

Examples

- $T = \Omega(d + 2^k)$ for binary trees for any algorithms;
- $T = O(d + 2^{\Theta(k)})$ for bounded degree graphs (BR);
- $T = \Omega(d + 2^k)$ for the path (BR)

State of the art

2 types of results

- *specialized deterministic* algorithms: ring, hypercube, complete graph, ...
- *universal randomized* algorithms with $\mathcal{M} = 0$: random walk (R), biasest random walk (BR)

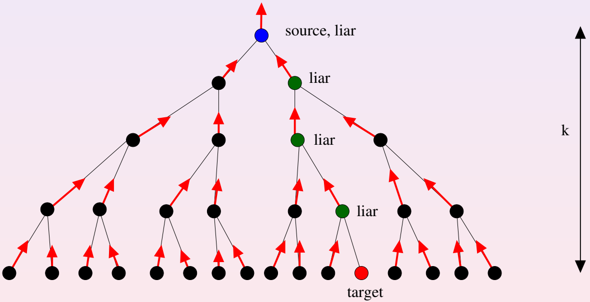
Examples

- $\mathcal{T} = \Omega(d + 2^k)$ for binary trees for any algorithms;
- $\mathcal{T} = O(d + 2^{\Theta(k)})$ for bounded degree graphs (BR);
- $\mathcal{T} = \Omega(d + 2^k)$ for the path (BR)

State of the art

Idea of the lower bound in trees

$\mathcal{T} = \Omega(d + 2^k)$ for binary trees for any algorithms;



Everything is symmetric: $\Omega(2^k)$ leaves look the same

Contribution

Goal

Designing *universal* algorithms using the least amount of requirements :

- no hypothesis on ports/nodes labeling ...
- light mobile agents ($\mathcal{M} = O(\log n)$)
- no knowledge on k.

New results

New algorithm: using the moody walk or R/A and $\mathcal{M} = O(\log k)$

- $\mathbb{E}(\mathcal{T}) = 2d + O(k^5)$ for the path
- $\mathbb{E}(\mathcal{T}) = O(k^3 \log^3 n)$ for some expanders (random regular)

Contribution

Goal

Designing *universal* algorithms using the least amount of requirements :

- no hypothesis on ports/nodes labeling ...
- light mobile agents ($\mathcal{M} = O(\log n)$)
- no knowledge on k.

New results

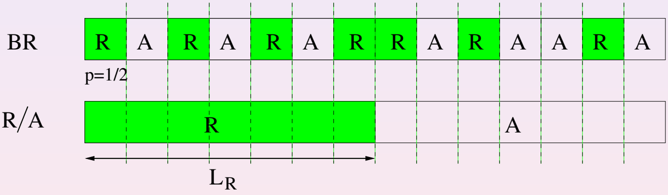
New algorithm: using the moody walk or R/A and $\mathcal{M} = O(\log k)$

- $\mathbb{E}(\mathcal{T}) = 2d + O(k^5)$ for the path
- $\mathbb{E}(\mathcal{T}) = O(k^3 \log^3 n)$ for some expanders (random regular)

Biasest random walk VS moody walk

Algorithm BR

- With probability p, follow the advice (A);
- Otherwise, the agent chooses a neighbor at random (R).



Algorithm R/A $[L_R, L_A]$

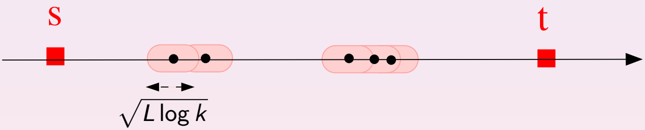
Keep on alternating Algorithm **R** for L_R hops and Algorithm **A** for L_A hops.

Ring or path

Gain: r.v X - distance reduction to t in one iteration.

Phase R during L steps

- With probability $1 - 2/k^c$, $|X_R| < \sqrt{L \log k}$;
- With probability $1 - 2/e$, $|X_R| < \sqrt{L}$.



Algorithm R/A[L, L]

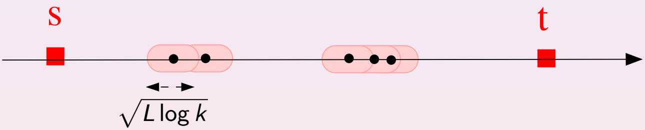
- Safe area: whp. $X_A > L - \sqrt{L \log k}$.
- Dangerous area: connected component of nodes at distance at most $\sqrt{L \log k}$ from liars.

Ring or path

Gain: r.v X - distance reduction to t in one iteration.

Phase R during L steps

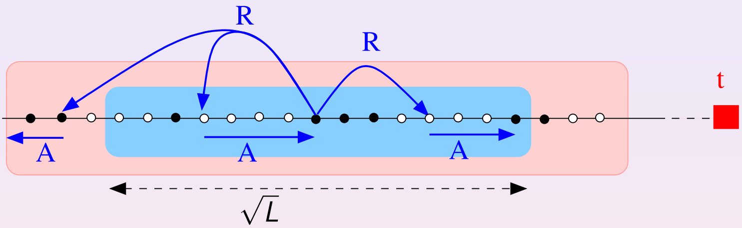
- With probability $1 - 2/k^c$, $|X_R| < \sqrt{L \log k}$;
- With probability $1 - 2/e$, $|X_R| < \sqrt{L}$.



Algorithm R/A[L, L]

- **Safe area**: whp. $X_A > L - \sqrt{L \log k}$.
- **Dangerous area**: connected component of nodes at distance at most $\sqrt{L \log k}$ from liars.

Path



Phase A (dangerous area)

1. $-k \leq X_A < 0$ with prob. $O(k/\sqrt{L})$;
2. $X_A \geq \sqrt{2L}/k - 1$ with prob. $\Theta(1)$;
3. $\mathbb{E}(X) = \mathbb{E}(X_A)$ and $\mathbb{E}(X_A) \geq \frac{\sqrt{L}}{40k} - 1$.

Performance of Moody walk on Path

With knowledge of k

For $L > 32k^3$, R/A[L,L] finds the target in
$\mathcal{T} = 2d + O(Lk^2 \log k) + o(d)$ with probability $1 - \Theta(1/k^{c-3})$,
$\mathcal{M} = O(\log k)$

Without knowledge of k

Iterating R/A$[2^{3i}, 2^{3i}]$, increasing i every $i2^{2i}$ phases:
$\mathcal{T} = 2d + O(k^5 \log k) + o(d)$ with probability $1 - \Theta(1/k^{c-3})$

Remainder: BR: $\mathcal{T} = \Theta(d + 2^k)$ for the path, $\mathcal{M} = 0$

Expanders

Graphs of high expansion are useful

- For any set S of nodes, the neighborhood of S is of size $\Omega(|S|)$
- P2P networks, error-correcting codes, small-world

A family of expanders: random Δ-regular graphs

- **Diameter**$=\log_{\Delta-1} n + \log_{\Delta-1} \log n + D$ (Bollobas, De la Vega 1982)
- **fast mixing**: Every node is reached with probability $\Theta(1/n)$ following a random walk of length $8\frac{\log n}{\log(\Delta/4)}$ (Friedman 2003, Cooper 2005)
- **Local Tree-like**: There is a constant $c = 1/2$ such that the neighborhood at distance $c \log_{\Delta-1} n$ is a tree (Cooper-Frieze-Radzik 2008)

Expanders

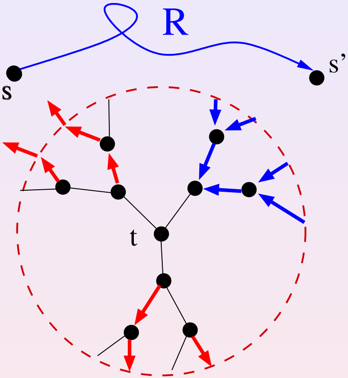
Graphs of high expansion are useful

- For any set S of nodes, the neighborhood of S is of size $\Omega(|S|)$
- P2P networks, error-correcting codes, small-world

A family of expanders: random Δ-regular graphs

- **Diameter**=$\log_{\Delta\text{-}1} n + \log_{\Delta\text{-}1} \log n + D$ (Bollobas, De la Vega 1982)
- **fast mixing**: Every node is reached with probability $\Theta(1/n)$ following a random walk of length $8\frac{\log n}{\log(\Delta/4)}$ (Friedman 2003, Cooper 2005)
- **Local Tree-like**: There is a constant $c \equiv 1/2$ such that the neighborhood at distance $c \log_{\Delta\text{-}1} n$ is a tree (Cooper-Frieze-Radzik 2008)

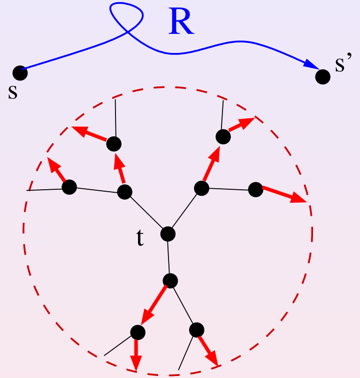
R/A/E with knowledge



R/A/E

- Beat the adversary ? Starting from a random source
- With prob. $\Theta(1)$, only truthtellers are encountered during phase A
- With prob. $\Theta(1)$, the target is "close" to the current position (phase E: BFS exploration)

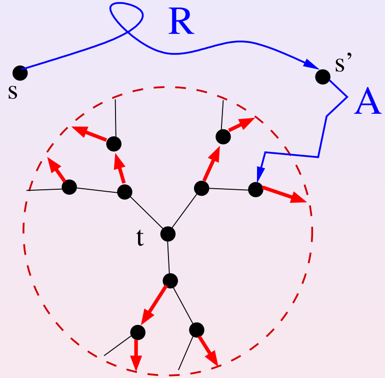
R/A/E with knowledge



R/A/E

- Beat the adversary ? Starting from a random source
- With prob. $\Theta(1)$, only truthtellers are encountered during phase A
- With prob. $\Theta(1)$, the target is "close" to the current position (phase E: BFS exploration)

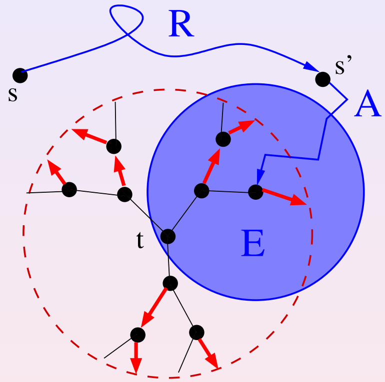
R/A/E with knowledge



R/A/E

- Beat the adversary ? Starting from a random source
- With prob. $\Theta(1)$, only truthtellers are encountered during phase A
- With prob. $\Theta(1)$, the target is "close" to the current position (phase E: BFS exploration)

R/A/E with knowledge



R/A/E

- Beat the adversary ? Starting from a random source
- With prob. $\Theta(1)$, only truthtellers are encountered during phase A
- With prob. $\Theta(1)$, the target is "close" to the current position (**phase E**: BFS exploration)

Toward R/A without knowledge of n and k

R/A/E WITH knowledge of n, k and diameter

- **Phase R**: $L_R = 8\frac{\log n}{\log(\Delta/4)}$ steps
- **Phase A**: $L_A = \log_{\Delta\text{-}1} n - 1 - \log_{\Delta\text{-}1} k$ steps
- **Phase E**: radius $r_E = \log_{\Delta\text{-}1} k + \log_{\Delta\text{-}1} \log n + D + 1$

Theorem: In a random Δ-regular graph, the target is found in $O(k \log n)$ steps with probability $\Theta(1)$.

R/A$[L_R + r_E, L_A]$ (Simulation of phase E)

In phase R, a target at distance r_E is reached with probability $\Omega(\frac{1}{k \log n})$.

Theorem: In a random Δ-regular graph, the target is reached in $O(k^2 \log^2 n)$ steps with probability $\Theta(1)$.

Toward R/A without knowledge of n and k

R/A/E WITH knowledge of n, k and diameter

- **Phase R**: $L_R = 8\frac{\log n}{\log(\Delta/4)}$ steps
- **Phase A**: $L_A = \log_{\Delta\text{-}1} n - 1 - \log_{\Delta\text{-}1} k$ steps
- **Phase E**: radius $r_E = \log_{\Delta\text{-}1} k + \log_{\Delta\text{-}1} \log n + D + 1$

Theorem: In a random Δ-regular graph, the target is found in $O(k \log n)$ steps with probability $\Theta(1)$.

R/A$[L_R + r_E, L_A]$ (Simulation of phase E)

In phase R, a target at distance r_E is reached with probability $\Omega(\frac{1}{k \log n})$.
Theorem: In a random Δ-regular graph, the target is reached in $O(k^2 \log^2 n)$ steps with probability $\Theta(1)$.

R/A/E without knowledge of n, k and diameter

R/A without knowledge

- $j \leftarrow 1$;
- $i = 1..j$, Run R/A[$100i, i$]
- $j++$

Theorem: In a random Δ-regular graph, the target is found within $O(c^3 k^3 \log^3 n)$ steps with probability $1 - 1/2^{\Omega(c)}$.

To sum up (without knowledge of n and k)

Strategy	Mean search time	
	Path	random Δ-regular
BR [$p < 1/2$]	$2^{\Omega(d)}$	$\Theta(\min\{(\Delta-1)^k, n^{\Theta(1)}\})$
BR [$p = 1/2$]	$\Omega(dn)$	$\Omega(\log^2_{\Delta\text{-}1} n)$
BR [$p > 1/2$]	$\Theta(d + 2^{\Theta(k)})$	$n^{\Theta(1)}$
R/A*	$2d + k^{\Theta(1)} + o(d)$	$O(k^3 \log^3 n)$
Lower B.*		$\Omega(\min\{(\Delta-1)^k, (\Delta-1)^d, \log_{\Delta\text{-}1} n\})$

*No hypothesis on the in-ports is assumed for the upper bounds but the lower bound assumes one.

What you need to retain

Do not blindy follow the advice but find the good proportions !!

Challenges

- Other graphs family: D-dimensionnal grids ?
- With less memory ?
- R/A$[n^3, 0]$ universal: How to parameter ?

What you need to retain

Do not blindy follow the advice but find the good proportions !!

Challenges

- Other graphs family: D-dimensionnal grids ?
- With less memory ?
- $\mathrm{R/A}[n^3, 0]$ universal: How to parameter ?