The expanding search ratio of a graph

Spyros Angelopoulos
CNRS, Sorbonne Universités, UPMC

Christoph Dürr
CNRS, Sorbonne Universités, UPMC

Thomas Lidbetter
London School of Economics → Rutgers

Find these slides at goo.gl/Y6V8AM

TU Berlin Seminar
Okt 2016
Outline

• Introduction
• Warmup: the cow-path problem
• The deterministic expanding search ratio on general graphs
• The randomized expanding search ratio on trees
• The randomized expanding search ratio on star graphs
Search problems

- You know: the search space
- You don’t know: where the hidden item is
- Decide: how to explore the search space as efficiently as possible
- Many Applications
2 search models

- **A search strategy:** is an ordering \( \pi \) on vertices with \( \pi(1) = \text{origin} \ O \)

- **Pathwise search:** connect \( \pi(i) \) to \( \pi(i-1) \)

- **Expanding search:** connect \( \pi(i) \) to the closest among \( \pi(1), \ldots, \pi(i-1) \)
a motivation for expanding search

- **You know:** A single mine is set in some crossing of a road network
- **You don’t know:** where
- Demining a road segment takes some time
- Walking along demined roads takes no time
- In what order should you proceed?
2 cost models

- Given search strategy $\pi$: every vertex $v$ is visited at some time $T(v)$ (= search time)

- Average search time:
  $$\min_{\pi} \frac{1}{|V \setminus O|} \sum_{v \in V \setminus O} T(v)$$

- Search ratio:
  $$\min_{\pi} \max_{v \in V \setminus O} \frac{T(v)}{d(O, v)}$$

- Randomized variant: distribution over search strategies that minimizes expected cost
Some previous work on graphs

<table>
<thead>
<tr>
<th>Pathwise search</th>
<th>Average search time</th>
<th>Search ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>also called minimum latency or traveling repairman problem</td>
<td>Blum Chalasani Coppersmith Pulleyblank Raghavan Sudan 1994</td>
</tr>
<tr>
<td>Expanding search</td>
<td>Alpern Lidbetter 2013</td>
<td>This work</td>
</tr>
</tbody>
</table>
### Results

<table>
<thead>
<tr>
<th></th>
<th>edge weighted graph</th>
<th>unweighted graph or edge weighted tree</th>
<th>edge weighted star graph</th>
</tr>
</thead>
</table>
| **Deterministic search ratio** | **NP-hard**  
**5.55 approximation** | polynomial                              |                          |
| **Randomized search ratio**    | 1.25 approximation                              | polynomial | [Condon et al'2009]     |
Outline

• Introduction

• Warmup: the cow-path problem

• The deterministic expanding search ratio on general graphs

• The randomized expanding search ratio on trees

• The randomized expanding search ratio on star graphs
cow path problem: definition

- there is juicy grass on the other side of the fence
- cow is at position 0, fence has an opening at position $x$ with $|x| \geq 1$ (sign($x$) is unknown to the cow)
- doubling ALG: walk to $+1, -2, +4, -8, +16, -32, \ldots$
- competitive ratio: = distance walked to opening / $|x|$
cow path problem: analysis

- worst case: \(|x| = 2^i + \epsilon\)

- cost of ALG:
  \[2(1+2+4+\ldots+2^{i+1}) + 2^i + \epsilon\]
  \[= 2(2^{i+2} - 1) + 2^i + \epsilon\]
  \[< 8 \cdot 2^i + 2^i + \epsilon\]
  \[< 9 \cdot \text{OPT}\]
Outline

- Introduction
- Warmup: the cow-path problem
- The deterministic expanding search ratio on general graphs
- The randomized expanding search ratio on trees
- The randomized expanding search ratio on star graphs
Get some intuition

- Suppose we give ourself an acceptable upper bound on the search ratio we want to attain
- Then we have a deadline (=UB on ratio times distance) for visiting each vertex

Don’t expand simply along a shortest path tree

Don’t expand simply along a minimum spanning tree
NP-hardness

- Computing the optimal search ratio is NP-hard
- Reduction from 3-SAT
- There are clause vertices, variable vertices, literal vertices, origin O, single vertex P
- If there is a satisfying assignment then there is an expanding search of ratio
  \[1 + \frac{2}{3} (\#\text{variables} + \#\text{clauses})\]
  and vice-versa
A 5.55 approximation

Algorithm

- sol = []
- For doubling radii d (1, 2, 4, …) do
  - let $B_d$ be the ball of vertices v with distance at most d from O
  - let $\hat{G}_d$ be a ln(4) approximation of the minimum Steiner tree of $B_d$ (Opt = $G_d$)
- append to sol an arbitrary expanding search of $\hat{G}_d$ (omitting edges useless for connectivity)
A 5.55 approximation

**Algorithm**
- $sol = []$
- For doubling radii $d (1,2,4,...)$ do
  - let $B_d$ be the ball of vertices $v$ with distance at most $d$ from $O$
  - let $\hat{G}_d$ be a $\ln(4)$ approximation of the minimum Steiner tree of $B_d$ ($Opt = G_d$)
  - append to $sol$ an arbitrary expanding search of $\hat{G}_d$ (omitting edges useless for connectivity)

**Analysis**
- Consider worst target $v$ is at level $d$
- Search ratio
  - $= w(\text{first tree hitting } v) / \text{dist}(O,v)$
  - $\leq \sum_{k=1,2,4,...d} w(\hat{G}_k) / \frac{1}{2}d$
  - $\leq \ln(4) \sum_{k=1,2,4,...d} w(G_k) / \frac{1}{2}d$
  - $\leq \ln(4) \text{ OptRatio } \sum_{k=1,2,4,...d} k / \frac{1}{2}d$
  - $\leq 4 \ln(4) \text{ OptRatio}$
A 5.55 approximation

Algorithm

- sol = []
- For doubling radii d (1, 2, 4, ...) do
  - let $B_d$ be the ball of vertices v with distance at most d from O
  - let $\hat{G}_d$ be a ln(4) approximation of the minimum Steiner tree of $B_d$ (Opt = $G_d$)
- append to sol an arbitrary expanding search of $\hat{G}_d$ (omitting edges useless for connectivity)

Key argument

- Minimum Steiner tree $G_d$ for $B_d$
- Consider Optimal expanding search
- and its first tree T that covers $B_d$
- then $w(T) \leq \text{OptRatio} \cdot d$
- and $w(T) \geq w(G_d)$
Outline

• Introduction
• Warmup: the cow-path problem
• The deterministic expanding search ratio on general graphs
• The randomized expanding search ratio on trees
• The randomized expanding search ratio on star graphs
A 5/4 approximation of the randomized search ratio

- **Theorem:** \( \text{ALG} \leq \frac{5}{4} \text{OPT} + 1 \)
  
  for \( \text{OPT} = \) optimal expected normalized search ratio 
  
  \( \text{ALG} = \) expected normalized search ratio of the...

- **Randomized deepening algorithm**

  - \( \text{sol} = [ ] \)
  
  - \( \text{T} = \{ O \} \)
  
  - For \( i = 0,1,2 \ldots \) do:
    
    - choose \( x_i \) uniformly at random in \( [2^{i-1}, 2^i] \)
    
    - add to \( \text{sol} \) all vertices with \( \text{dist}[T,v] \leq x_i \) in depth first search order with random child order
    
    - \( \text{T} = \text{sol} \)
Outline

• Introduction
• Warmup: the cow-path problem
• The deterministic expanding search ratio on general graphs
• The randomized expanding search ratio on trees
• The randomized expanding search ratio on star graphs
Expanding search on star graphs

Opening times of the boxes are known. What probability distribution on box orders should we follow to minimize the time to find the star?
Example

<table>
<thead>
<tr>
<th></th>
<th>probability</th>
<th></th>
<th>expected search time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>4/5</td>
<td>1/5</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Search order</th>
<th>AB</th>
<th>BA</th>
</tr>
</thead>
<tbody>
<tr>
<td>target at A</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>4/5 · 1 + 1/5 · 3 = 7/5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>target at B</td>
<td>3/2</td>
<td>2/2</td>
</tr>
<tr>
<td>4/5 · 3/2 + 1/5 · 2/2 = 14/10</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Results

- We recursively define a strategy that is optimal in some case
- Condon et al. 2009 gave an $O(n^2)$ algorithm to compute an optimal distribution
Thank you