Sequential observer selection for source localization

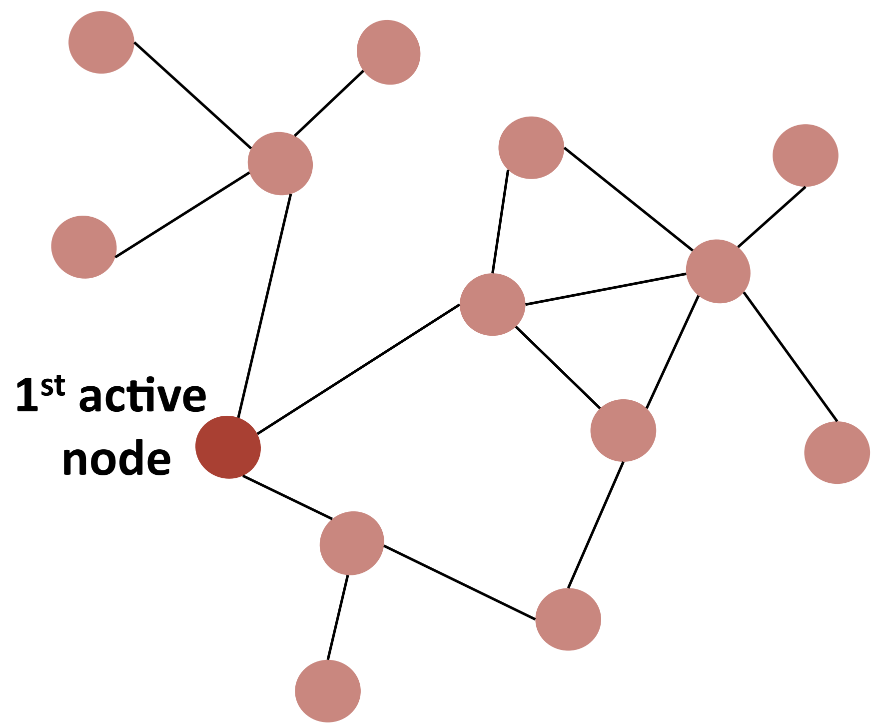
Sabina Zejnilović *†, João Gomes† and Bruno Sinopoli*

†Instituto Superior Técnico, Universidade de Lisboa, Portugal

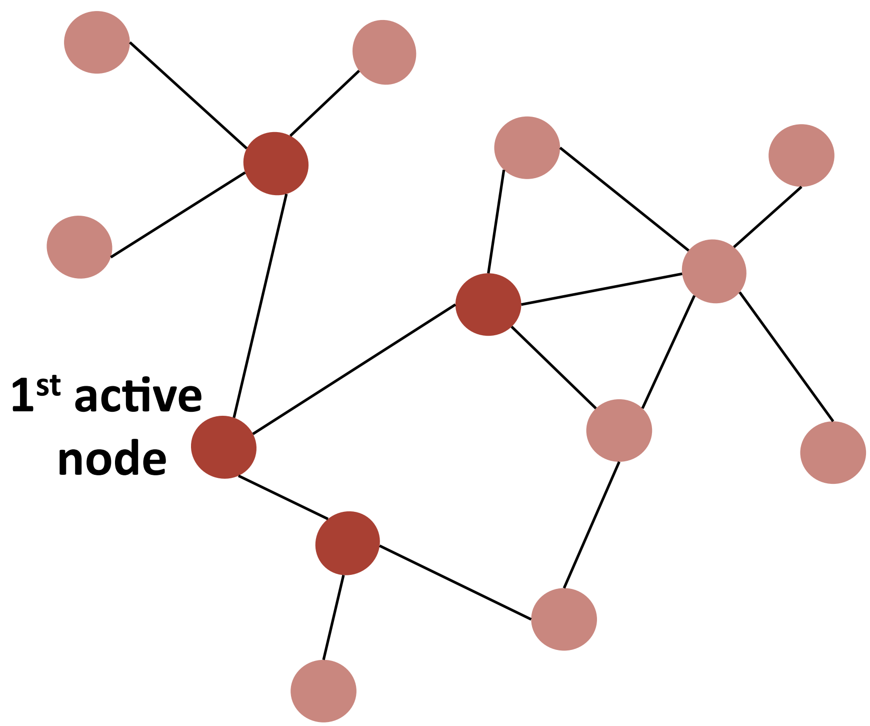
*Carnegie Mellon University, Pittsburgh, PA

Information and Communication Technologies Institute
Carnegie Mellon | PORTUGAL
AN INTERNATIONAL PARTNERSHIP

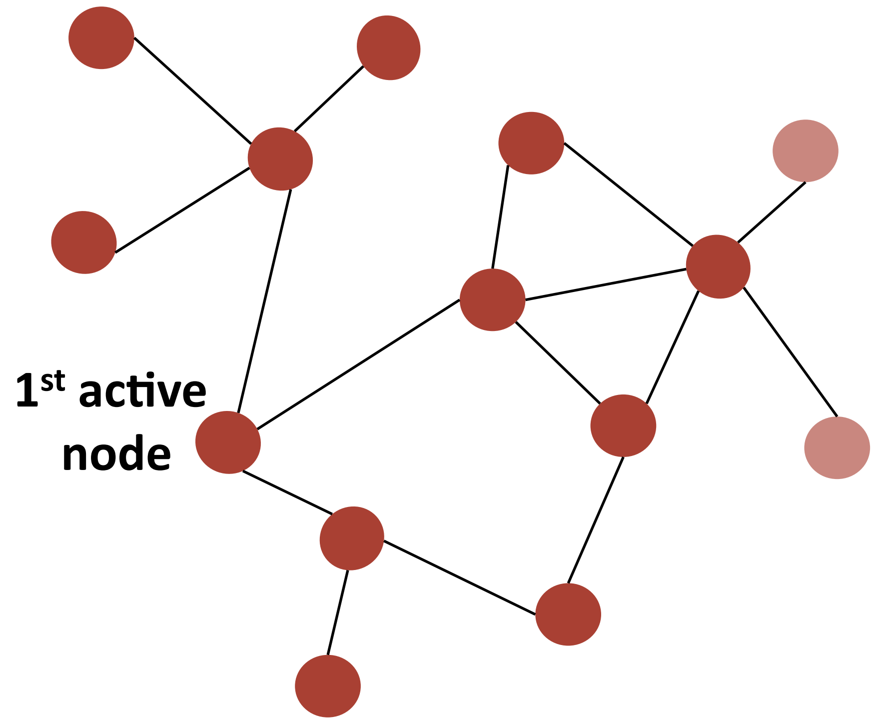
Different phenomena modeled as network diffusion



Different phenomena modeled as network diffusion

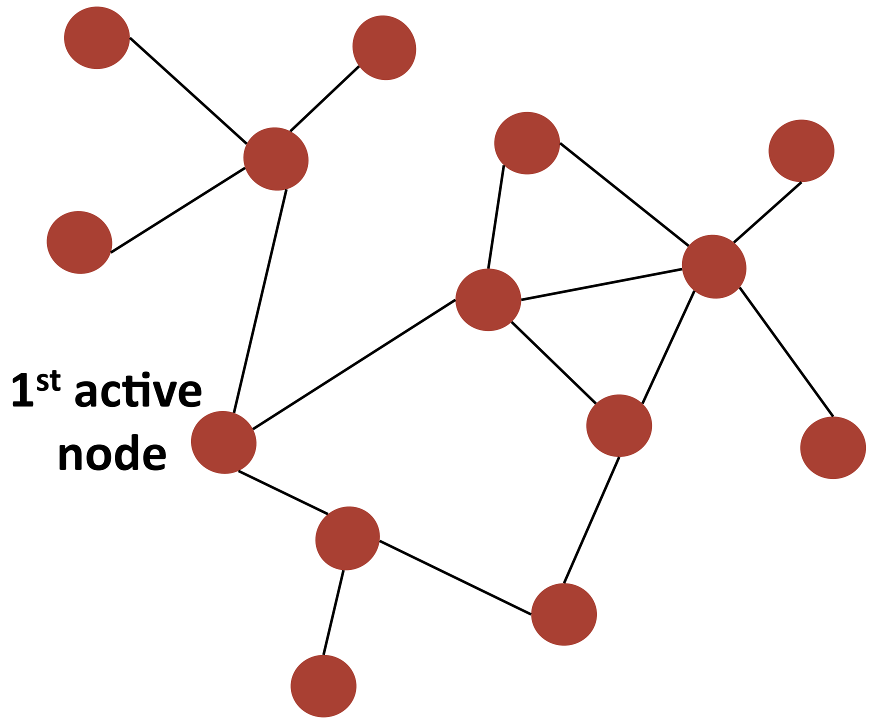


Different phenomena modeled as network diffusion



Different phenomena modeled as network diffusion

- *Propagation of a disease in human population*
- *Dissemination of information in a social network*
- *Spreading of a computer virus in a communication network*

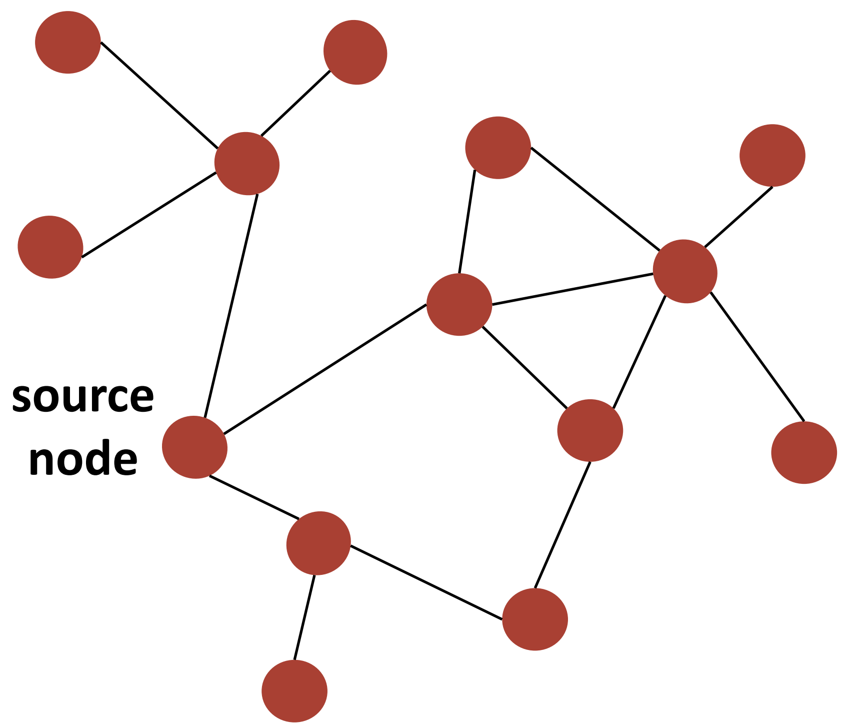


Different phenomena modeled as network diffusion

- *Propagation of a disease in human population*
- *Dissemination of information in a social network*
- *Spreading of a computer virus in a communication network*

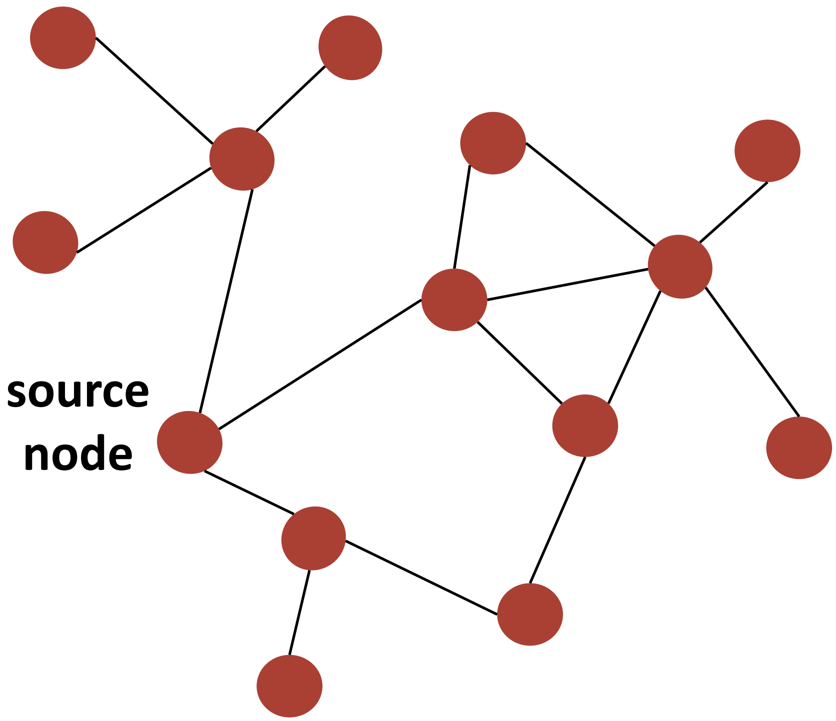
Source of diffusion

- *Patient zero*
- *Trendsetter*
- *1st infected computer*



Different phenomena modeled as network diffusion

- *Propagation of a disease in human population*
- *Dissemination of information in a social network*
- *Spreading of a computer virus in a communication network*



Source of diffusion

- *Patient zero*
- *Trendsetter*
- *1st infected computer*

Limited access to state of the network nodes

- ***Network size***
- ***Privacy issues***
- ***Cost of observation***

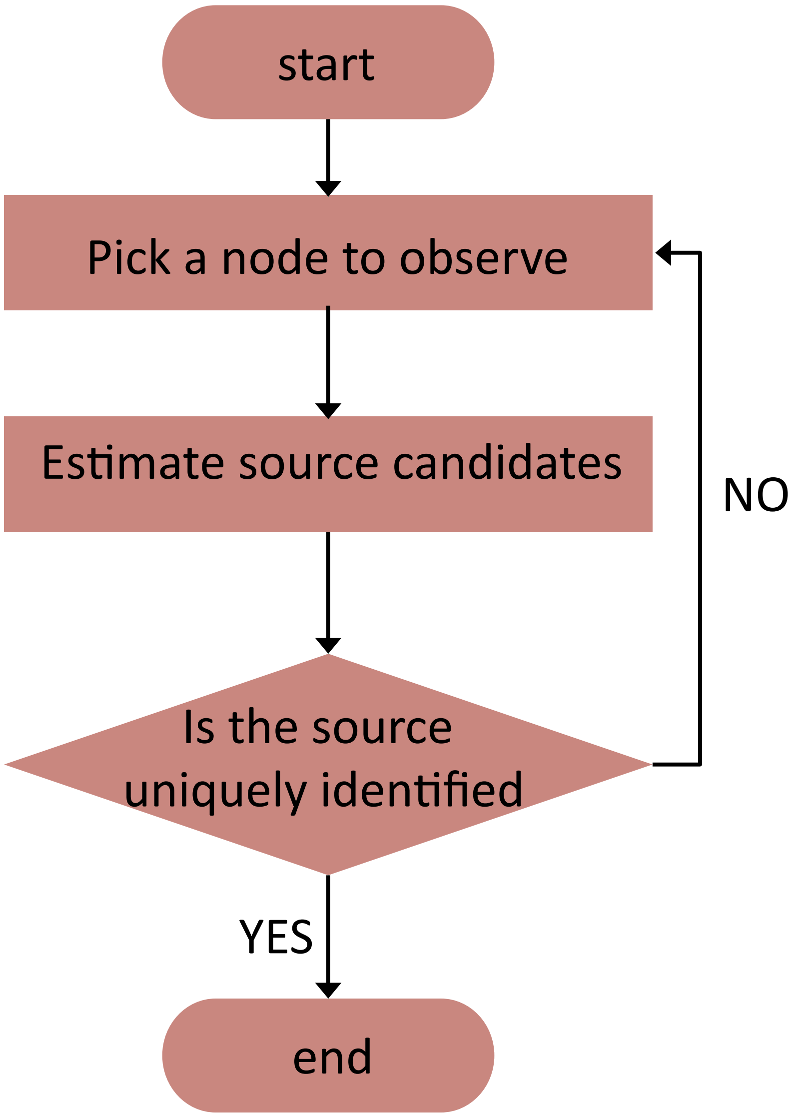
Research on network diffusion

- parameters of network diffusion
- source localization with or without timestamps of infection
- strategies for selection of the nodes that are observed: offline, mostly simulation-based

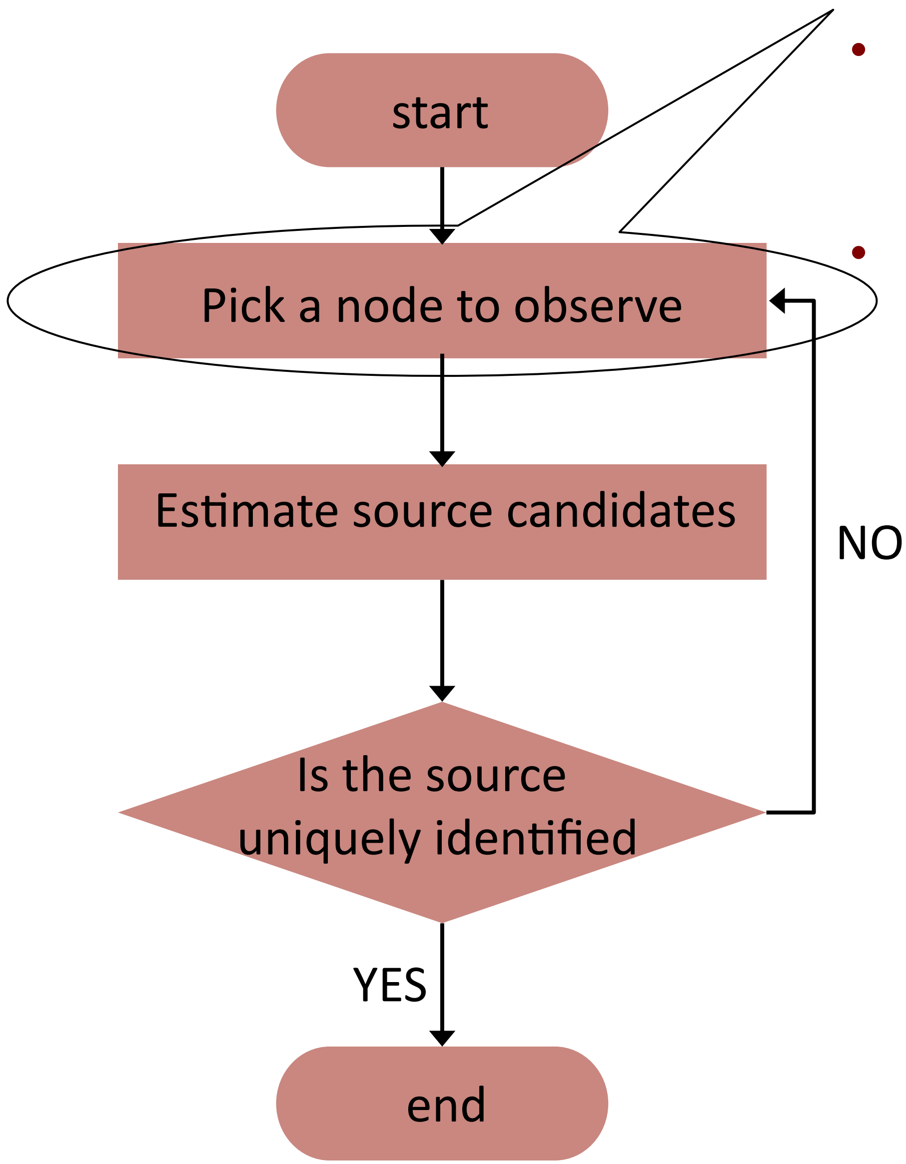
Contributions

- *sequential* dynamic selection
- *theoretical analysis* of the optimal selection strategy
    - show it is combinatorial problem even with the simplified assumptions
    - provide optimal solution
    - derive efficient approximation, yet with guarantees
    - gain insight for more complex assumptions

Sequential identification of the source



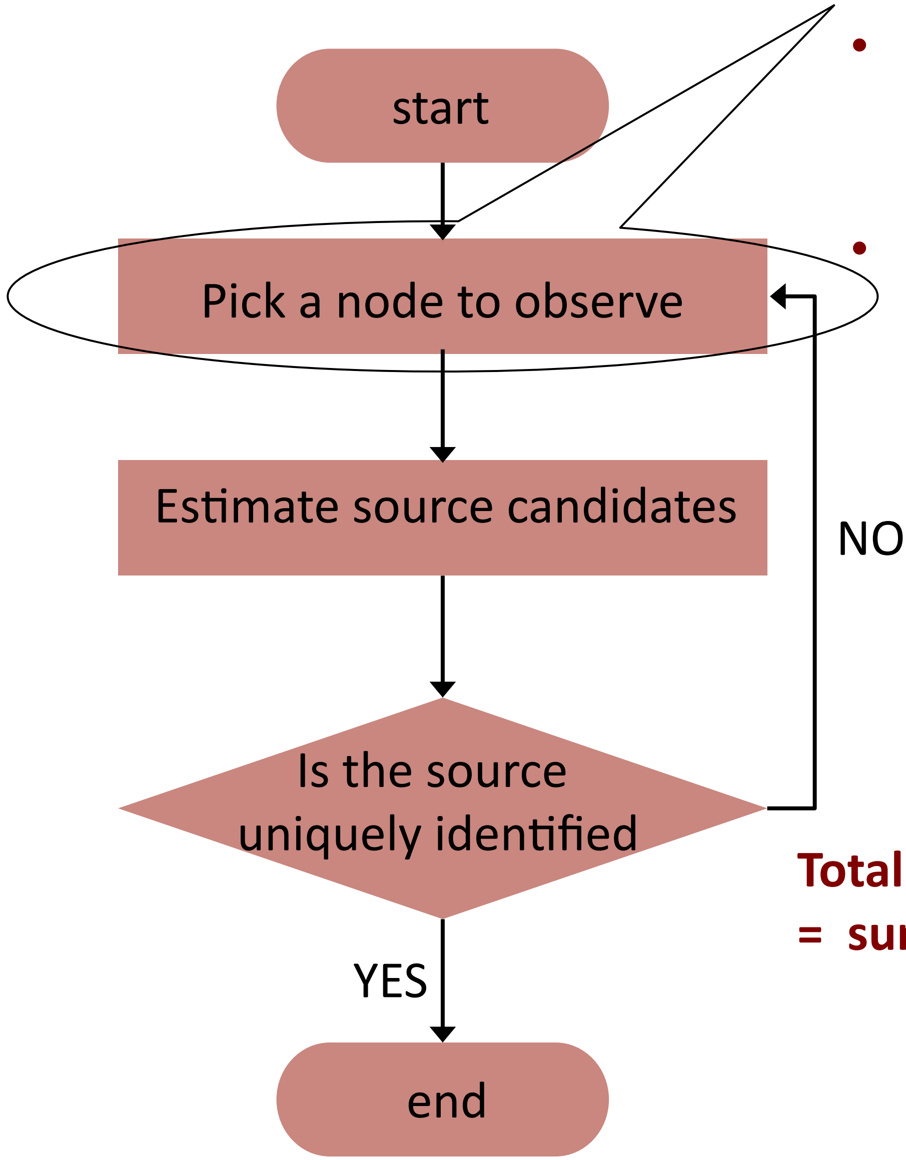
Sequential identification of the source



Network node

- **Information about the source through its observation: timestamp of infection**
- **Cost of observation**

Sequential identification of the source

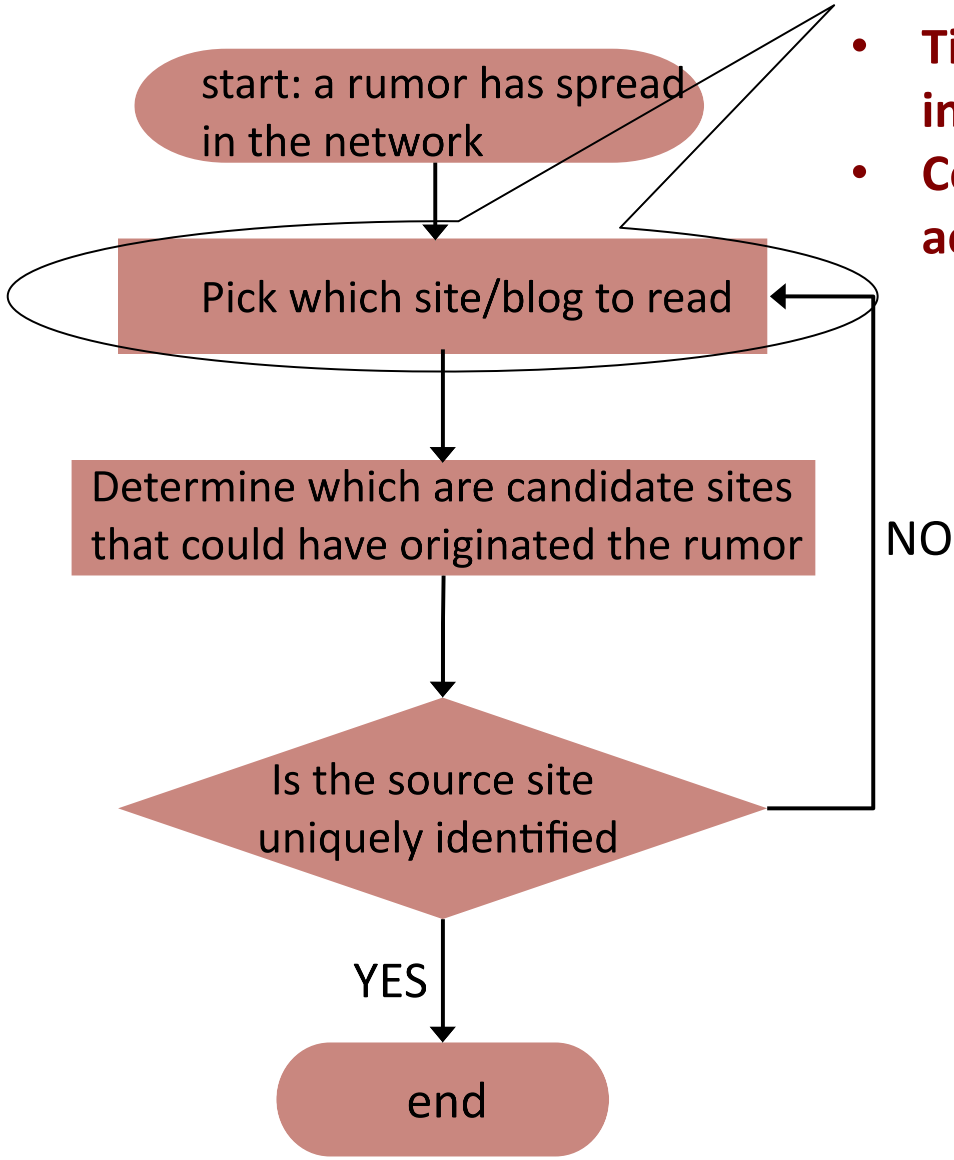


Network node

- **Information about the source through its observation: timestamp of infection**
- **Cost of observation**

Total cost of source identification
= sum of costs of observers

Sequential identification of the rumor source



Each site/blog has

- **Timestamp when it posted information**
- **Cost (time, effort) of accessing the site**

Problem statement

1. **Find a selection strategy such that the source can be unambiguously localized with the smallest total cost.**

Problem statement

1. Find a selection strategy such that the source can be unambiguously localized with the smallest total cost.

2. **For a fixed number of observer nodes find a selection strategy that would result with the smallest number of source candidates.**

Proposed approaches

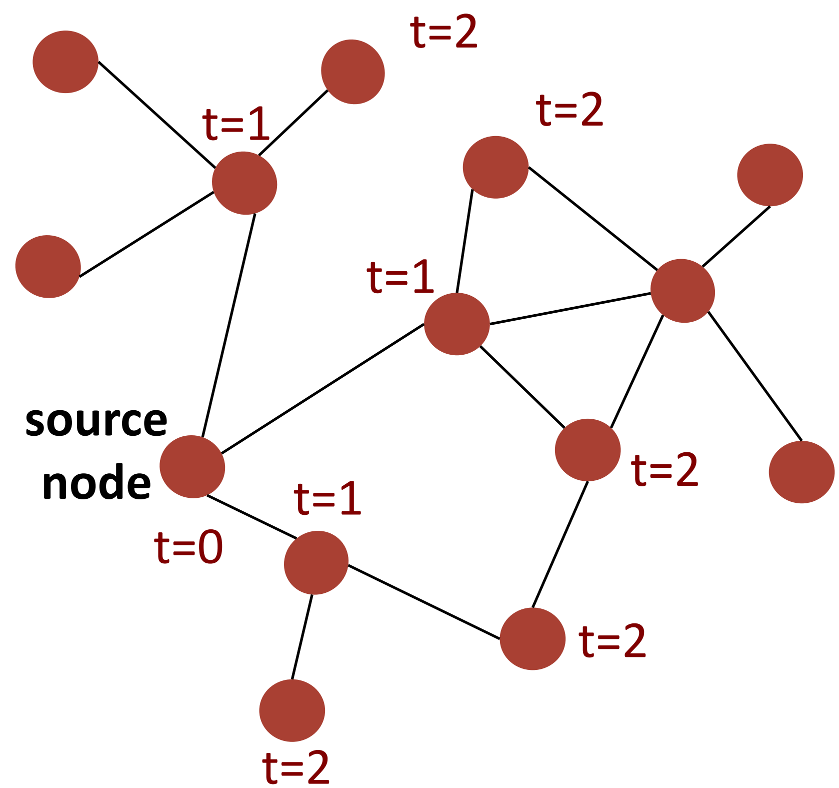
Approach	Optimality	Efficiency
Dynamic programming	✔	
Greedy		✔

Proposed approaches

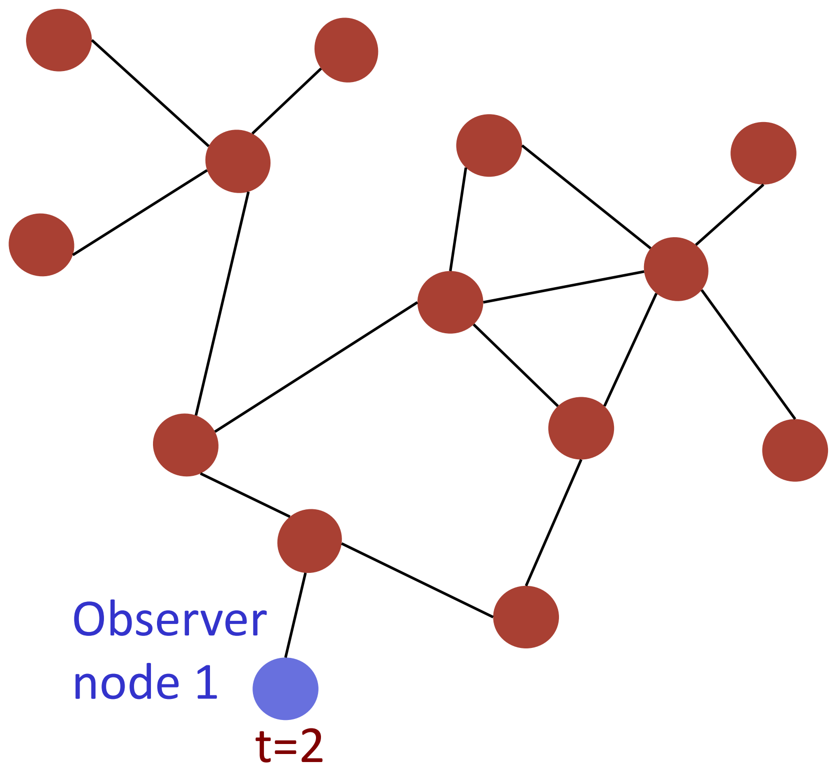
Approach	**Optimality**	**Efficiency**	**Performance guarantees**
Dynamic programming	✔		✔
Greedy (adaptive submodularity)		✔	✔

A simple model of network diffusion

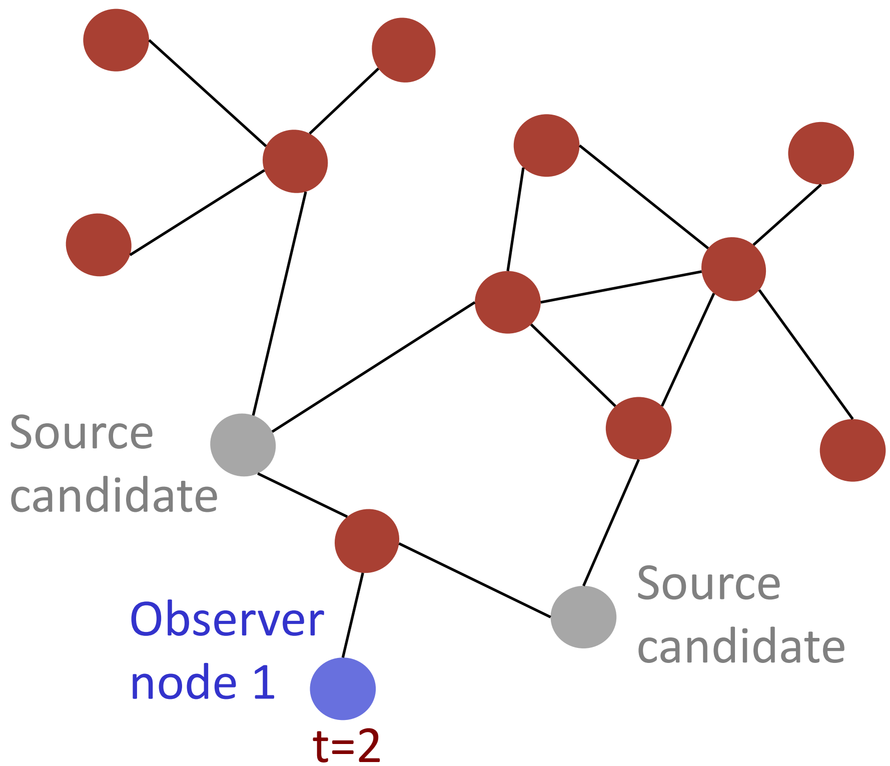
- Single source node
- Nodes either susceptible or infected
- Nodes infected at time *t* infect neighbors with probability 1 at next time step *t+1*
- Times of infections deterministic: distance to the source



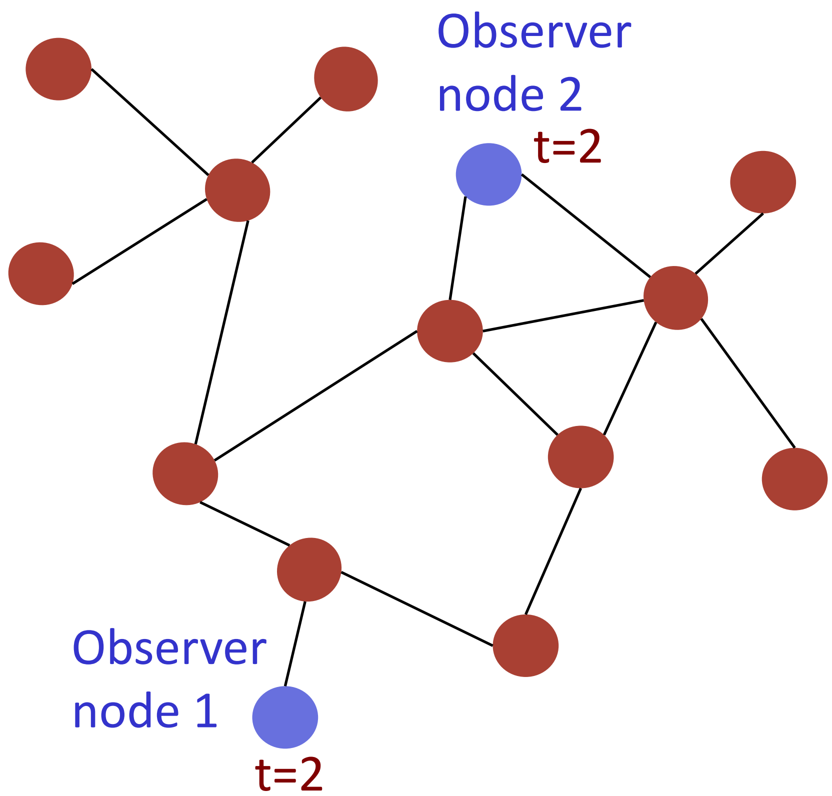
Source is localized based on the infection times of observed nodes



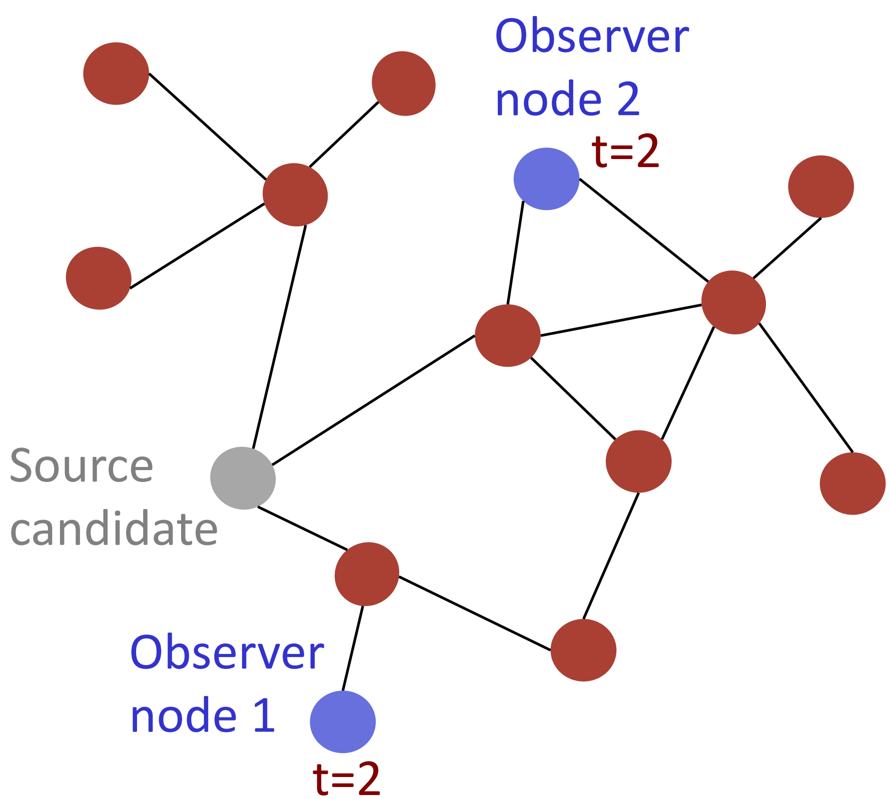
Source is localized based on the infection times of observed nodes



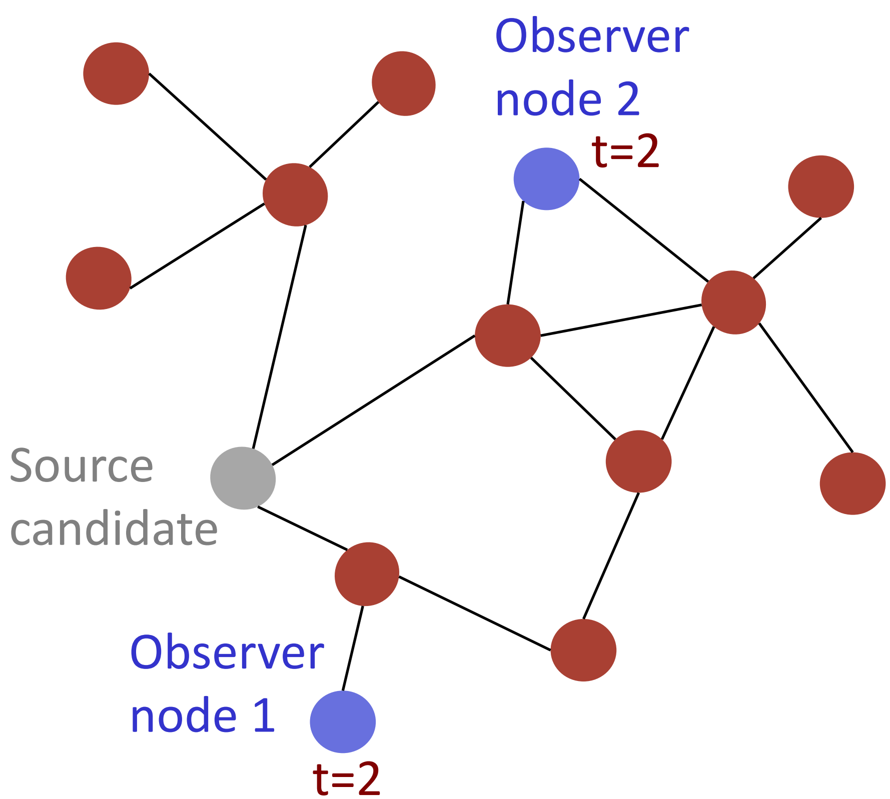
Source is localized based on the infection times of observed nodes



Source is localized based on the infection times of observed nodes



Source is localized based on the infection times of observed nodes



Total cost incurred = cost of observing Observer 1 + cost of observing Observer 2

Problem 1 formulation

Find a selection strategy π such that the source can be unambiguously localized with the smallest cost.

$$\min_{\pi} \mathbb{E}_s \left[c \left(O \left(\pi \right) \right) \right]$$

$$\text{subject to } \boldsymbol{d}(O(\pi), s) \neq \boldsymbol{d}(O(\pi), i), \forall s \in V, s \neq i,$$

Problem 1 formulation

Find a selection strategy π such that the source can be unambiguously localized with the smallest cost.

$$\min_{\pi} \mathbb{E}_s \left[c\left(O\left(\pi\right)\right)\right]$$

$$\text{subject to } \boldsymbol{d}(O\left(\pi\right), s) \neq \boldsymbol{d}(O\left(\pi\right), i), \forall s \in V, s \neq i,$$

average taken over all possible sources

cost c incurred by observing a subset of nodes O chosen by strategy ***π***

Problem 1 formulation

Find a selection strategy π such that the source can be unambiguously localized with the smallest cost.

$$\min_{\pi} \mathbb{E}_s \left[c \left(O \left(\pi \right) \right) \right]$$

$$\text{subject to } \boldsymbol{d}(O\left(\pi\right), s) \neq \boldsymbol{d}(O\left(\pi\right), i), \forall s \in V, s \neq i,$$

while each source node has a unique identifier

Problem 2 formulation

For a fixed number of observer nodes T find a selection strategy that would result with the smallest number of source candidates.

$$\min_{\pi} \mathbb{E}_s |S(O(\pi))|$$
$$\text{subject to } |O(\pi)| \leq T$$

Problem 2 formulation

For a fixed number of observer nodes T find a selection strategy that would result with the smallest number of source candidates.

$$\min_{\pi} \mathbb{E}_s |S(O(\pi))|$$
$$\text{subject to } |O(\pi)| \leq T$$

average taken over all possible sources

number of source candidates given by observing a subset of nodes O chosen by strategy $\boldsymbol{\pi}$

Problem 2 formulation

For a fixed number of observer nodes T find a selection strategy that would result with the smallest number of source candidates.

$$\min_{\pi} \mathbb{E}_s |S(O(\pi))|$$

$$\text{subject to } |O(\pi)| \leq T$$

while the number of observers is no more than T

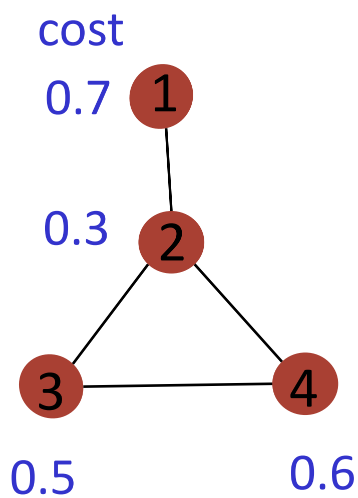
Solve both problems using dynamic programming with imperfect state knowledge

Problem is analyzed

- **backwards:** from the selection of the last observer to the selection of the first observer, one step at the time
- **offline:** considering all the possible sources, deriving what should be the best observer to select for possible observations

Example: Dynamic programming approach for Problem 1

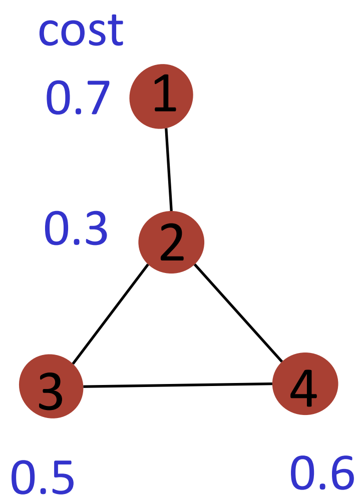
Analysis for cost incurred by selecting node 2 in the first step



distance	node 1	node 2	node 3	node 4
node 2	1	0	1	1

Example: Dynamic programming approach for Problem 1

Analysis for cost incurred by selecting node 2 in the first step

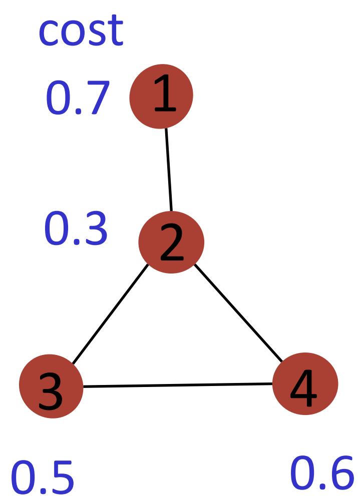


distance	node 1	node 2	node 3	node 4
node 2	1	0	1	1

- *source 1,3,4:* t=1, source candidates {1,3,4}, prob 3/4
- *source 2:* t=0, source candidates {2} , prob 1/4

Example: Dynamic programming approach for Problem 1

Analysis for cost incurred by selecting node 2 in the first step



distance	node 1	node 2	node 3	node 4
node 2	1	0	1	1

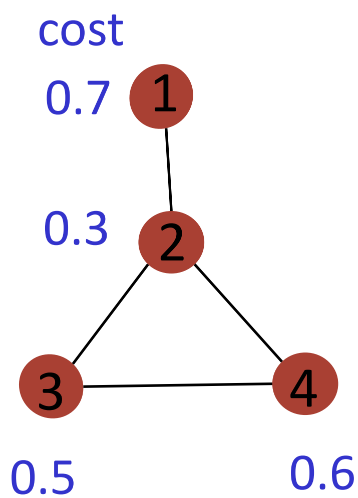
- *source 1,3,4*: t=1, source candidates {1,3,4}, prob 3/4
- *source 2*: t=0, source candidates {2} , prob 1/4

In the previous step, step 2, we have calculated

- cost-to-go of state {1,3,4} as 0.5
- cost-to-go of state {2} as 0

Example: Dynamic programming approach for Problem 1

Analysis for cost incurred by selecting node 2 in the first step



distance	node 1	node 2	node 3	node 4
node 2	1	0	1	1

- *source 1,3,4*: t=1, source candidates {1,3,4}, prob 3/4
- *source 2*: t=0, source candidates {2} , prob 1/4

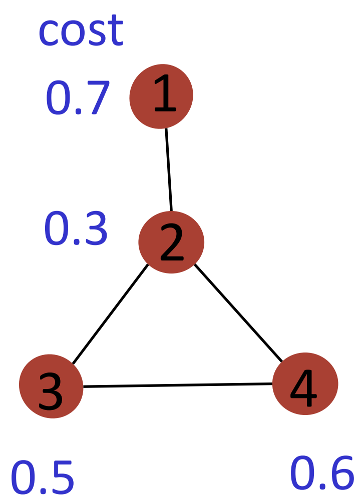
In the previous step, step 2, we have calculated

- cost-to-go of state {1,3,4} as 0.5
- cost-to-go of state {2} as 0

Cost-to-go of selecting node 2= cost(node 2)+(3/4 *0.5+1/4*0)

Example: Dynamic programming approach for Problem 1

Analysis for cost incurred by selecting node 2 in the first step



distance	node 1	node 2	node 3	node 4
node 2	1	0	1	1

- *source 1,3,4*: t=1, source candidates {1,3,4}, prob 3/4
- *source 2*: t=0, source candidates {2} , prob 1/4

In the previous step, step 2, we have calculated

- cost-to-go of state {1,3,4} as 0.5
- cost-to-go of state {2} as 0

Cost-to-go of selecting node 2= cost(node 2)+(3/4 *0.5+1/4*0)

Optimal node for step 1 is the node with the smallest cost-to-go.

Dynamic programming approach for Problem 1:

Selecting an optimal observer for an arbitrary time step k

$$\underset{\text{(observations)}}{\underset{\textbf{step k}}{\textbf{cost-to-go}}} = \underset{\text{observer o}}{\textbf{min}}\ \textbf{E}_{\textit{sources}} \left[\text{cost(o)} + \underset{\text{step k+1}}{\text{cost-to-go}}\ \text{(observations)} \right]$$

Dynamic programming approach for Problem 1:
Selecting an optimal observer for an arbitrary time step k

cost-to-go step k (observations) **=** **min** observer o **E** ***sources*** [cost(o) + cost-to-go (observations) step k+1]

Optimal observer step k (observations) **=** **arg min** *observer o* **cost-to-go step k** (observations)

Dynamic programming approach for Problem 1:
Selecting an optimal observer for an arbitrary time step k

$$\underset{\text{(observations)}}{\overset{\textbf{cost-to-go}}{\textbf{step k}}} = \min_{\text{observer } o} \mathbf{E}_{\textit{sources}} \left[\text{cost}(o) + \underset{\text{step k+1}}{\text{cost-to-go}} \text{(observations)} \right]$$

$$\underset{\text{(observations)}}{\textbf{Optimal observer step k}} = \underset{\textit{observer } o}{\textbf{arg min}} \underset{\text{(observations)}}{\overset{\textbf{cost-to-go}}{\textbf{step k}}}$$

Dynamic programming is optimal, but generally intractable

- combinatorial nature of the problem

Efficient approximation, but with performance guarantees

- In order to obtain guarantees we resort to *adaptive submodularity** :
  if an optimization problem has this property, greedy approach has guarantees

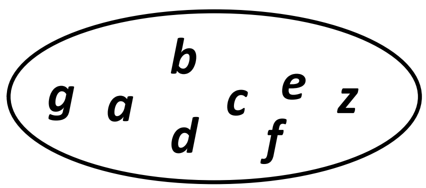
*D. Golovin, A. Krause, "*Adaptive Submodularity: Theory and Applications in Active Learning and Stochastic Optimization*", Journal of Artificial Intelligence Research, 2011

Efficient approximation, but with performance guarantees

- In order to obtain guarantees we resort to *adaptive submodularity** :
  if an optimization problem has this property, greedy approach has guarantees

Adaptive submodularity – generalization of diminishing returns

set of items,
unknown states



pick an item

b

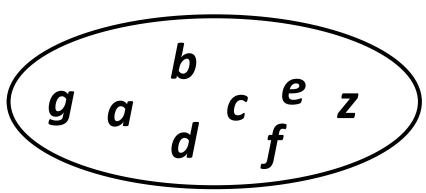
*D. Golovin, A. Krause, *"Adaptive Submodularity: Theory and Applications in Active Learning and Stochastic Optimization"*, Journal of Artificial Intelligence Research, 2011

Efficient approximation, but with performance guarantees

- In order to obtain guarantees we resort to *adaptive submodularity** :
  if an optimization problem has this property, greedy approach has guarantees

Adaptive submodularity

set of items,
unknown states



b

↓

learn the state
of the item

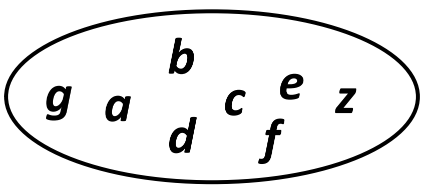
*D. Golovin, A. Krause, *"Adaptive Submodularity: Theory and Applications in Active Learning and Stochastic Optimization"*, Journal of Artificial Intelligence Research, 2011

Efficient approximation, but with performance guarantees

- In order to obtain guarantees we resort to *adaptive submodularity** :
  if an optimization problem has this property, greedy approach has guarantees

Adaptive submodularity

set of items,
unknown states



evaluate utility
function

b

↓

learn the state
of the item

↓

update our posterior
belief about the state
of the world

*D. Golovin, A. Krause, *"Adaptive Submodularity: Theory and Applications in Active Learning and Stochastic Optimization"*, Journal of Artificial Intelligence Research, 2011

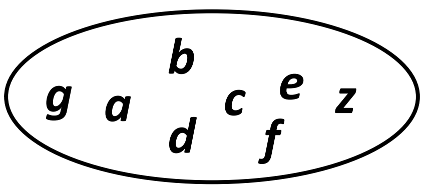
Efficient approximation, but with performance guarantees

- In order to obtain guarantees we resort to *adaptive submodularity** :
  if an optimization problem has this property, greedy approach has guarantees

Adaptive submodularity

set of items,
unknown states

b, f



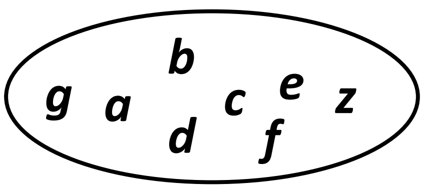
*D. Golovin, A. Krause, "*Adaptive Submodularity: Theory and Applications in Active Learning and Stochastic Optimization*", Journal of Artificial Intelligence Research, 2011

Efficient approximation, but with performance guarantees

- In order to obtain guarantees we resort to *adaptive submodularity** :
  if an optimization problem has this property, greedy approach has guarantees

Adaptive submodularity

set of items,
unknown states



evaluate utility
function

b, f

↓

learn the state
of the item

↓

update our posterior
belief about the state
of the world

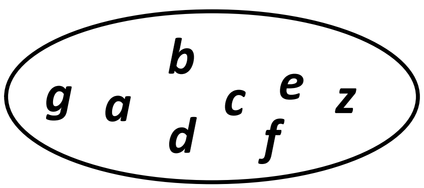
*D. Golovin, A. Krause, “*Adaptive Submodularity: Theory and Applications in Active Learning and Stochastic Optimization*”, Journal of Artificial Intelligence Research, 2011

Efficient approximation, but with performance guarantees

- In order to obtain guarantees we resort to *adaptive submodularity** :
  if an optimization problem has this property, greedy approach has guarantees

Adaptive submodularity

set of items,
unknown states



evaluate utility
function

b, f, z

↓

learn the state
of the item

↓

update our posterior
belief about the state
of the world

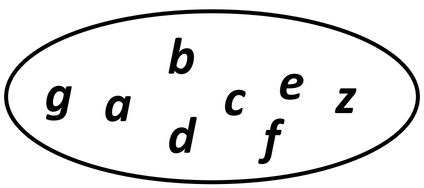
*D. Golovin, A. Krause, *"Adaptive Submodularity: Theory and Applications in Active Learning and Stochastic Optimization"*, Journal of Artificial Intelligence Research, 2011

Efficient approximation, but with performance guarantees

- In order to obtain guarantees we resort to *adaptive submodularity** :
  if an optimization problem has this property, greedy approach has guarantees

Adaptive submodularity

set of items,
unknown states



b, f, z

Shorter sequence

b, f, g, e, z

Longer sequence

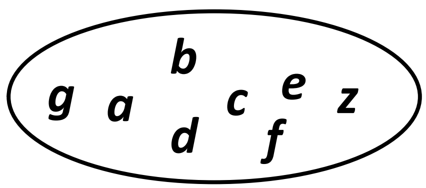
*D. Golovin, A. Krause, "*Adaptive Submodularity: Theory and Applications in Active Learning and Stochastic Optimization*", Journal of Artificial Intelligence Research, 2011

Efficient approximation, but with performance guarantees

- In order to obtain guarantees we resort to *adaptive submodularity** :
  if an optimization problem has this property, greedy approach has guarantees

Adaptive submodularity

set of items,
unknown states



expected increase in utility function after adding z ≥ *expected increase in utility function after adding z*

b, f, z

Shorter sequence

b, f, g, e, z

Longer sequence

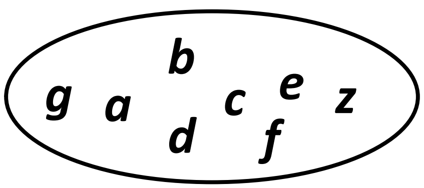
*D. Golovin, A. Krause, *"Adaptive Submodularity: Theory and Applications in Active Learning and Stochastic Optimization"*, Journal of Artificial Intelligence Research, 2011

Efficient approximation, but with performance guarantees

- In order to obtain guarantees we resort to *adaptive submodularity** :
  if an optimization problem has this property, greedy approach has guarantees

Adaptive submodularity

set of items,
unknown states



expected increase in utility function after adding z	≥	*expected increase in utility function after adding z*
b, f, z		***b, f, g, e, z***
Shorter sequence		Longer sequence

Expectation is taken with different posterior probability distributions

*D. Golovin, A. Krause, "*Adaptive Submodularity: Theory and Applications in Active Learning and Stochastic Optimization*", Journal of Artificial Intelligence Research, 2011

We can reformulate problems 1 and 2 such that they have adaptive submodularity property

- ***Introduce utility function f=N-|S(O)|: number of nodes that are not source candidates after observing O***
- ***Prove f=N-|S(O)| is adaptive monotone and adaptive submodular for uniform source prior***
- ***Obtain performance guarantees for greedy selection***

Initial formulation of problem 1

$$\min_{\pi} \mathbb{E}_s \left[c \left(O \left(\pi \right) \right) \right]$$

$$\text{subject to } \boldsymbol{d}(O(\pi), s) \neq \boldsymbol{d}(O(\pi), i), \forall s \in V, s \neq i,$$

Reformulation of problem 1 as Adaptive Stochastic Minimum Cost Cover

$$\min_{\pi} \; \mathbb{E}_s \left[c \left(O \left(\pi \right) \right) \right]$$

$$\text{subject to } N - \left| S \left(O \left(\pi \right) \right) \right| \geq N - 1, \forall s \in V, s \neq i$$

Initial formulation of problem 2

$$\min_{\pi} \mathbb{E}_s |S(O(\pi))|$$
$$\text{subject to } |O(\pi)| \leq T$$

Reformulate problem 2 as Adaptive Stochastic Maximization

$$\max_{\pi} \mathbb{E}_s [N - |S(O(\pi))|]$$
$$\text{subject to } |O(\pi)| \leq T.$$

Selection of the best observer at step k for the greedy approach

observer at step k = **arg max** (observer o) **1/c(o)** **E** *current source candidates* **[** decrease in the number of source candidates after selecting observer o **]**

$$\text{observer at step } k = \underset{\text{observer } o}{\arg\max} \; \frac{1}{c(o)} \; E_{\textit{current source candidates}} \left[\text{decrease in the number of source candidates after selecting observer } o \right]$$

Selection of the best observer at step k for the greedy approach

$$\text{observer at step } k = \underset{\text{observer } o}{\arg\max} \; \frac{1}{c(o)} \; E_{\textit{current source candidates}} \left[\text{decrease in the number of source candidates after selecting observer } o \right]$$

Performance guarantees

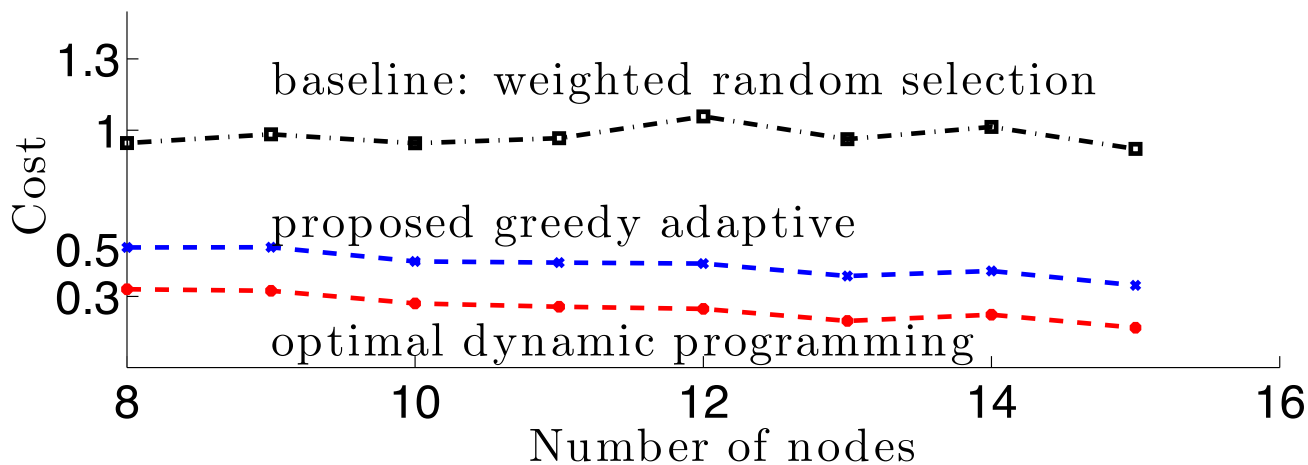
Problem 1. **cost by greedy ≤ optimal cost (log(N(N-1))+1)**

Problem 2. **# candidates by greedy ≤ optimal #candidates (1-1/e) +N/e**

Cost incurred by different approaches for solving problem 1

- Benchmark against the performance of a weighted random selection

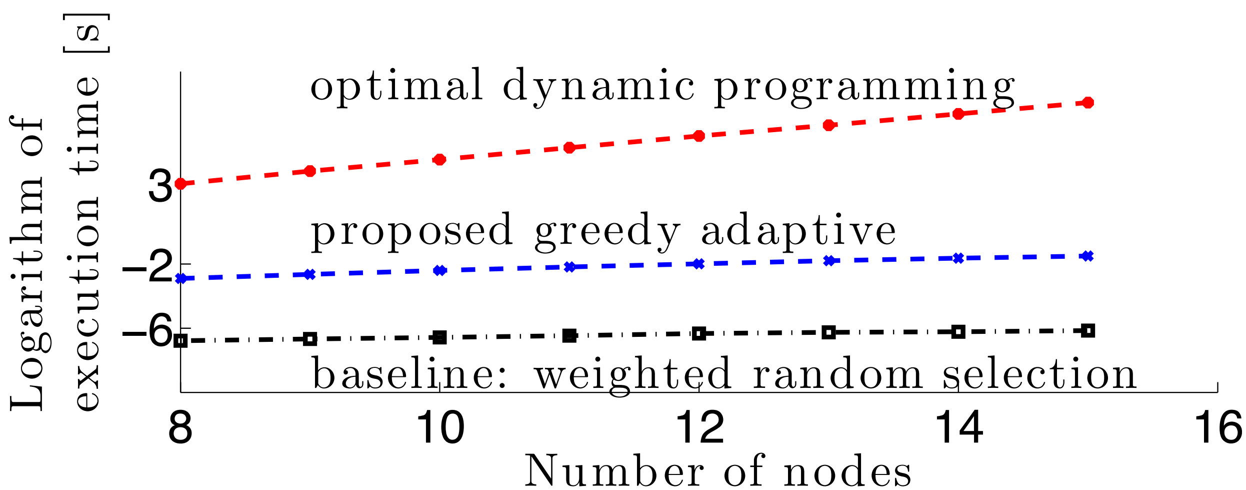
- 100 realizations of small-world networks
- uniform source prior
- node cost random uniform [0,1]



Time required by different approaches to solve problem 1

- Benchmark against the performance of a weighted random selection

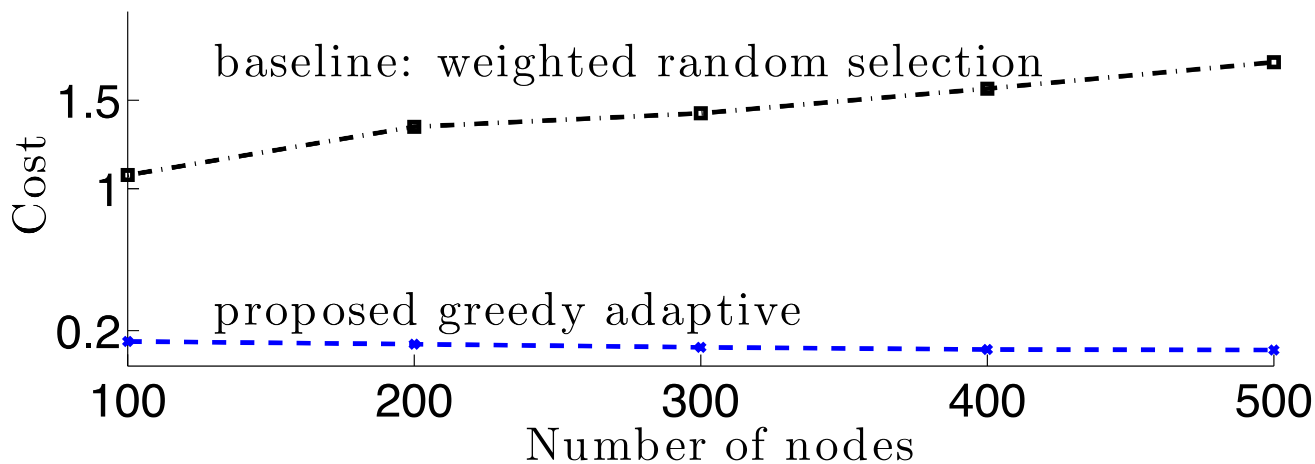
- 100 realizations of small-world networks
- uniform source prior
- node cost random uniform [0,1]



Cost incurred by approximate approaches for solving problem 1

- Benchmark against the performance of a weighted random selection

- 100 realizations of small-world networks
- uniform source prior
- node cost random uniform [0,1]



Conclusions and future work

- Formulated two problems:
  - minimize the cost for unambiguous source localization
  - minimize the number of source candidates after observing a prespecified number of nodes
- Solved problems optimally with stochastic dynamic programming
- Used adaptive submodularity to formulate a greedy algorithm with performance guarantees
- Future work: extend the model to stochastic propagation time