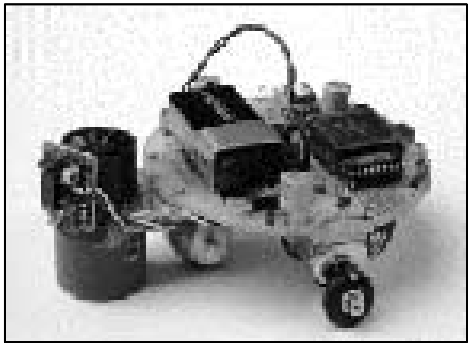
Tutorial on Ant Robotics

Sven Koenig
University of Southern California
skoenig@usc.edu

joint work with Jonas Svennebring, Boleslaw Szymanski, Yaxin Liu, and Craig Tovey



Overview

- Cheap robots
  - Limited capabilities
    - Computation capability
    - Sensing capability
    - Actuation capability
  - Groups of robots
    - Fault tolerance
    - Parallelism

Robomow


Cye

Roomba

Overview

- Ant robots
  - Robots with limited capabilities
  - Robots that leave information in the terrain
- Ant robots cannot use conventional planning methods. Rather, their behavior is driven by local interactions. This can result in very robust navigation.

Recommended special journal issue for further reading:
AMAI special issue on Ant Robotics edited by Wagner and Bruckstein

Overview

- Our motivating task is the one-time or repeated coverage of known or unknown terrain with single robots or teams of robots
  - Mine sweeping
  - Surveillance
  - Search-and-rescue
  - Guarding
  - Surface inspection

Overview

- This topic is a bit far out. However, …
- We will touch on different areas of AI and CS
  - Agent coordination (swarms)
  - Robotics (robot architectures, ant robots, sensor networks)
  - Search (real-time search)
  - Complexity analysis of graph algorithms
- We will see several good dissertation topics.

your idea here

Structure

- Motivation
- Real-time search
- Results on real-time search
- Application to ant robots and results
- Serious application: smart markers

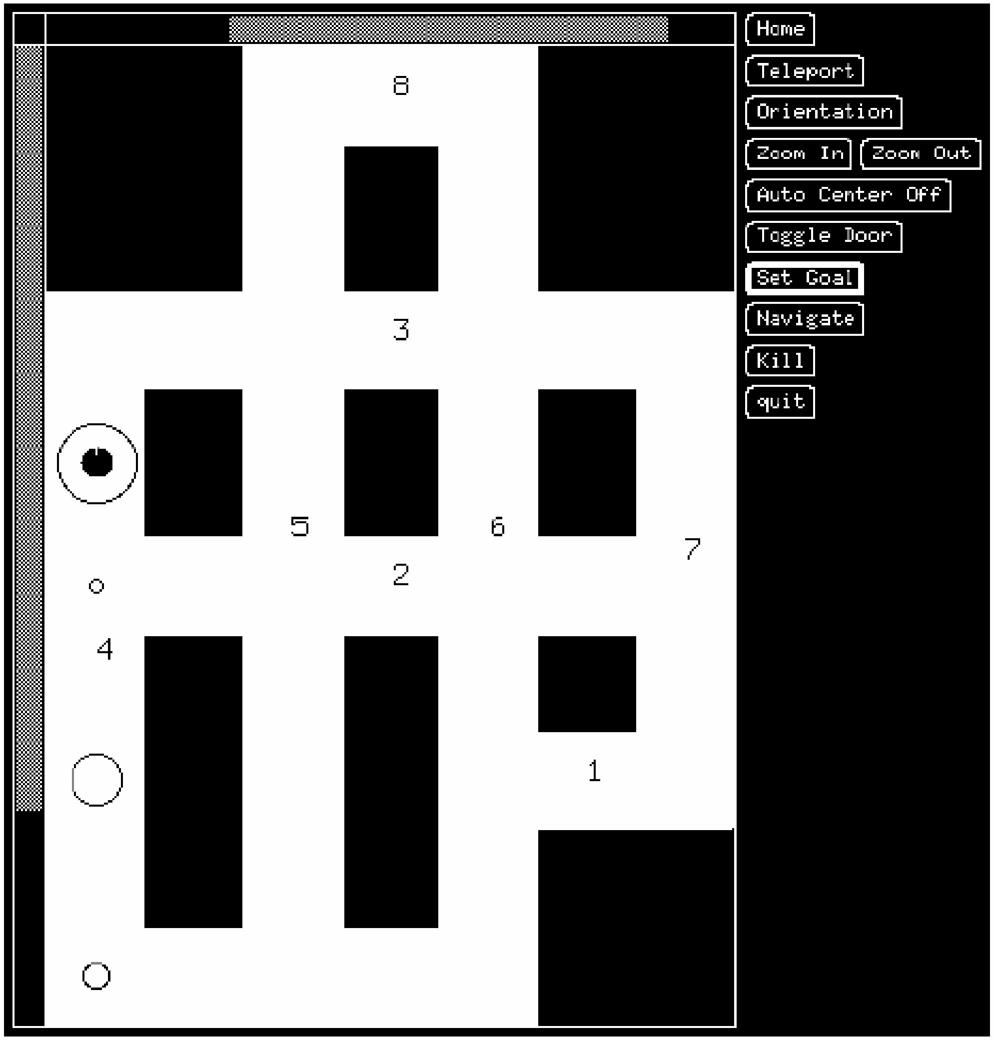
Motivating Toy Task

- Guarding a museum at night
  - Robots
    - Computation is slow
    - Sensing is noisy
    - Actuation is noisy
    - Robots can fail
  - Terrain
    - Terrain might be unknown initially
    - Terrain can change over time

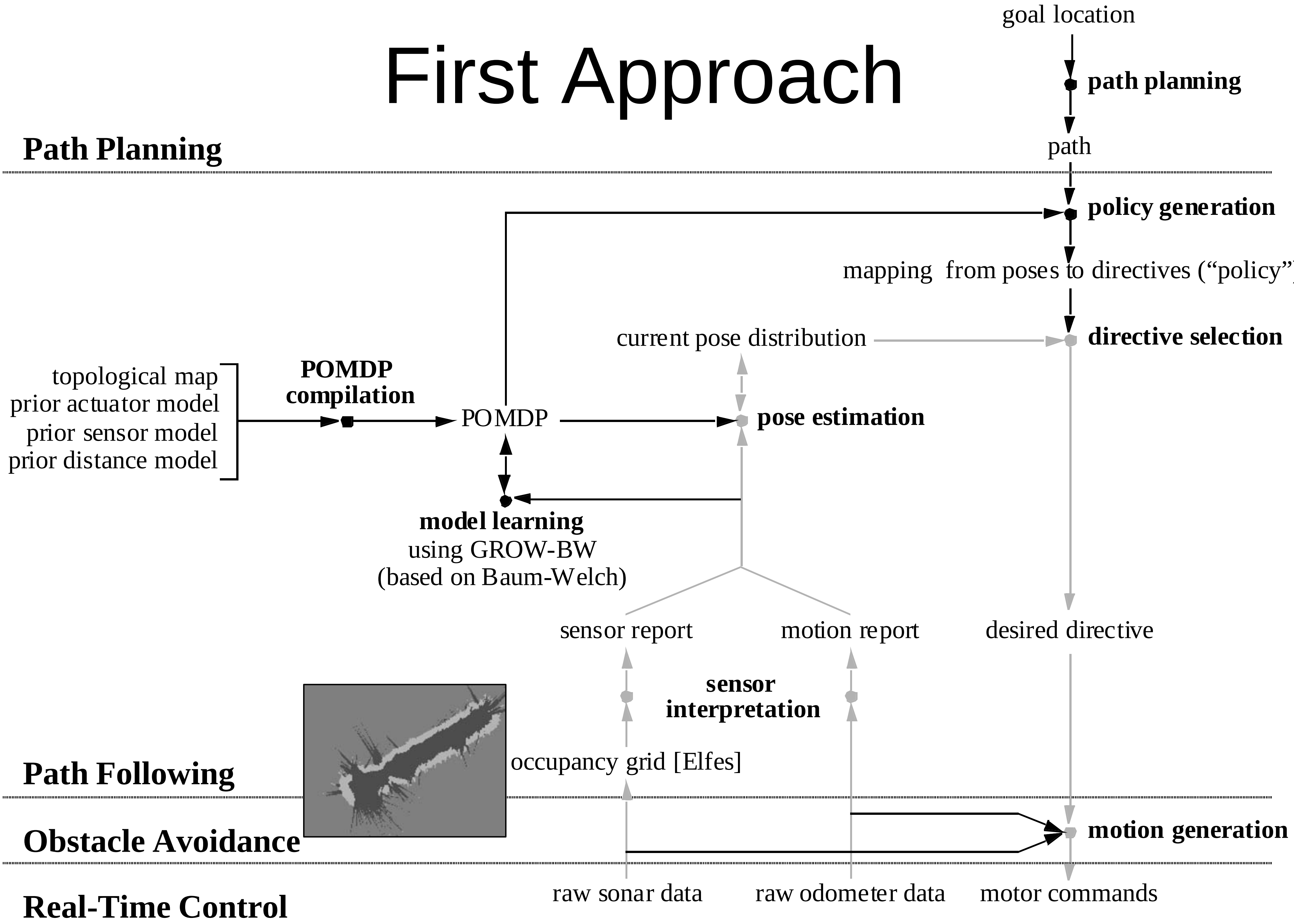
First Approach

- Good location estimates - e.g. probabilities
- Path planning – e.g. POMDPs
- Explicit coordination – e.g. auctions

First Approach



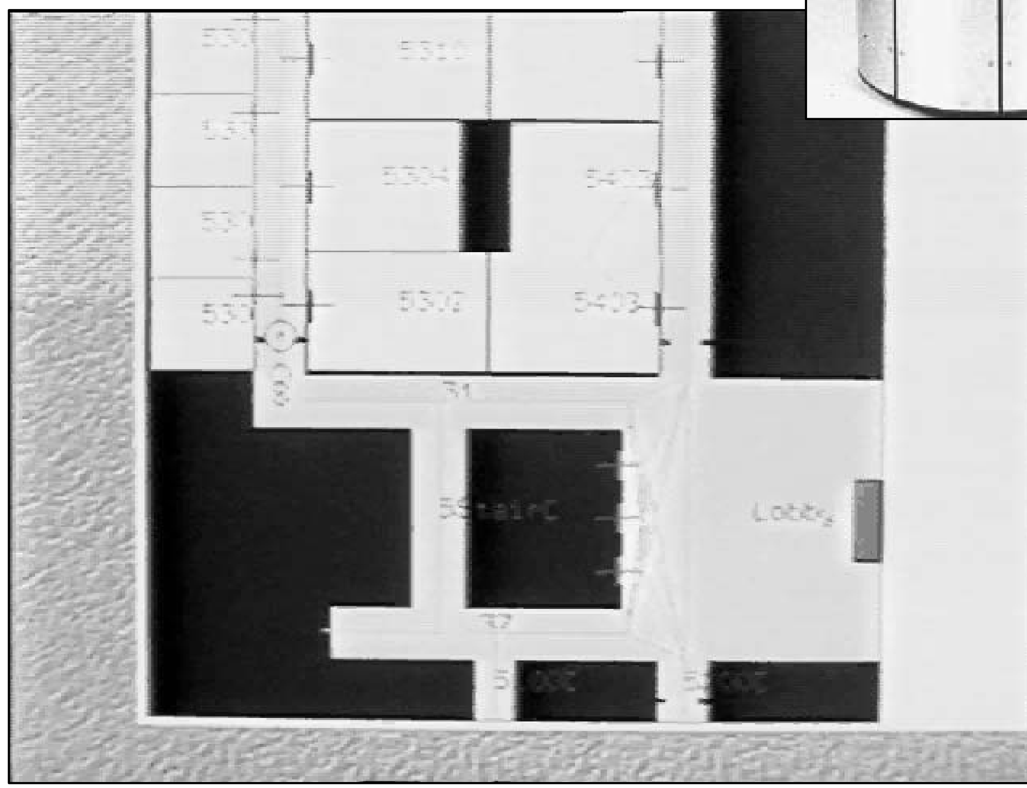
First Approach



First Approach

- Example: Xavier [Simmons and Koenig]

Xavier is a mobile robot at Carnegie Mellon University that received navigation requests from users worldwide via the World Wide Web and used POMDP-based navigation to travel more than 230 kilometers – an early used of POMP-based navigation which is now in wide-spread use.



First (Standard) Approach

- Good location estimates - e.g. probabilities
- Path planning – e.g. POMDPs
- Explicit coordination – e.g. auctions

- The standard approach
- Complex hardware and software

Recommended book for further reading:
Probabilistic Robotics, Thrun, Burgard and Fox, MIT Press

Second Approach

- No location estimates
- No planning
- No explicit coordination

- Not a standard approach at all
- Simpler hardware and software

Second Approach

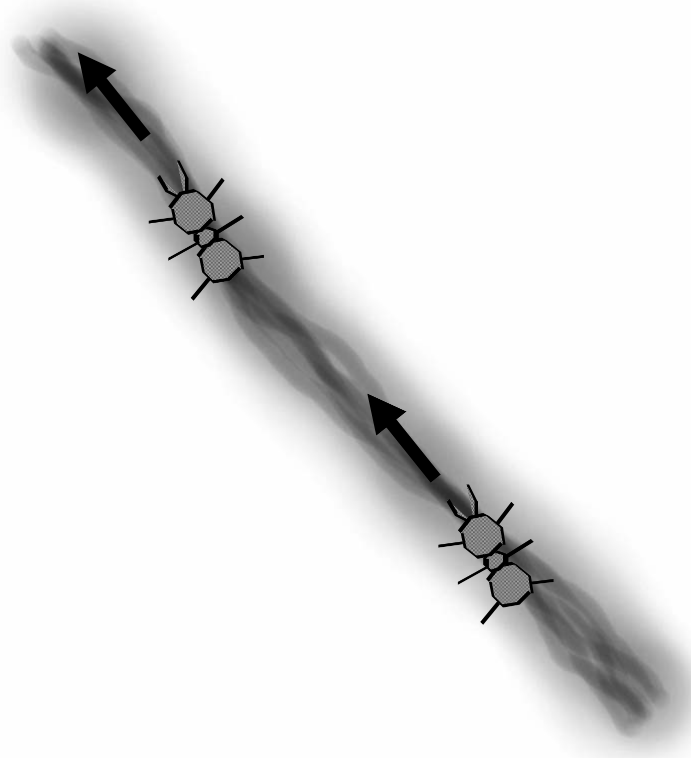
- No location estimates
- No planning
- No explicit coordination

- Random walk

Second Approach

- No location estimates
- No planning
- No explicit coordination

- Leaving trails in the terrain
  - Short-lived trails
    - Heat [Russell]
    - Odor [Russell et al.]
    - Alcohol [Sharpe et al.]



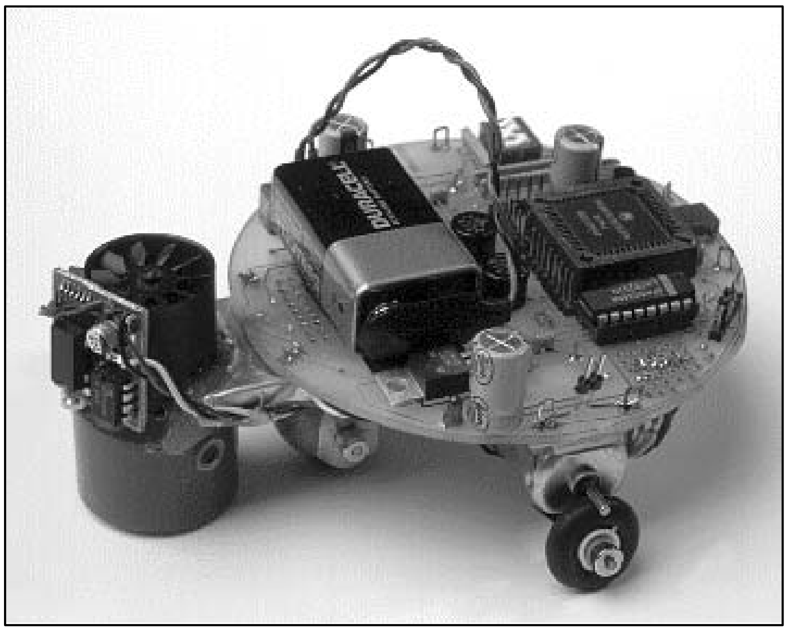
Second Approach

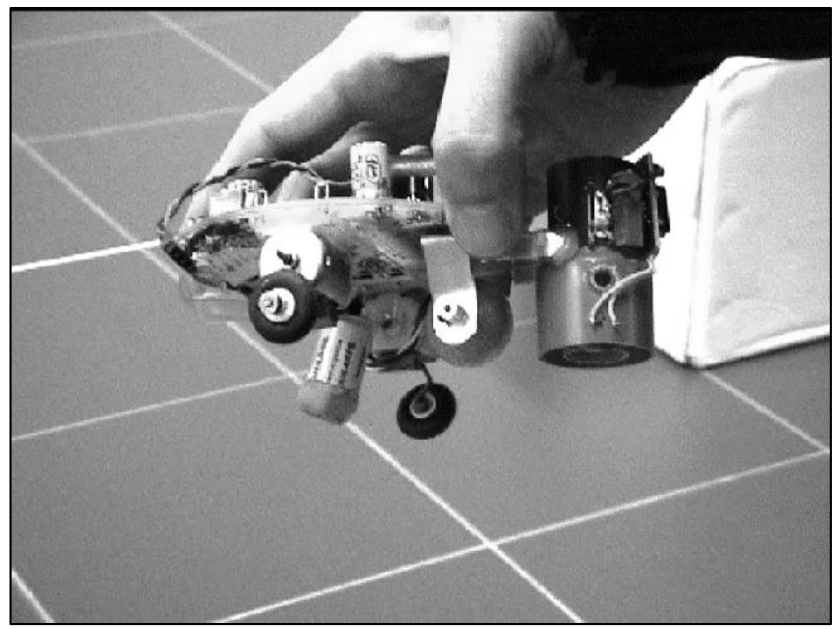
- Chemical sensing is a relatively new area of robotics with many interesting challenges and important applications.

Recommended book for further reading:
Odour Sensing for Mobile Robots, Russell, World Scientific

Second Approach

- Andrew Rusell's RAT robot lays a camphor trail and then follows it back to its starting point.

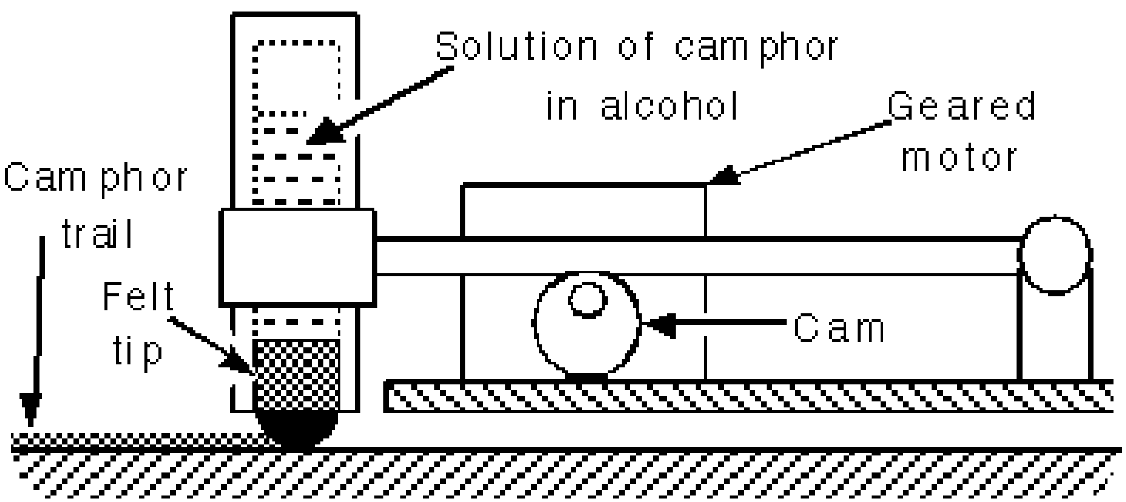


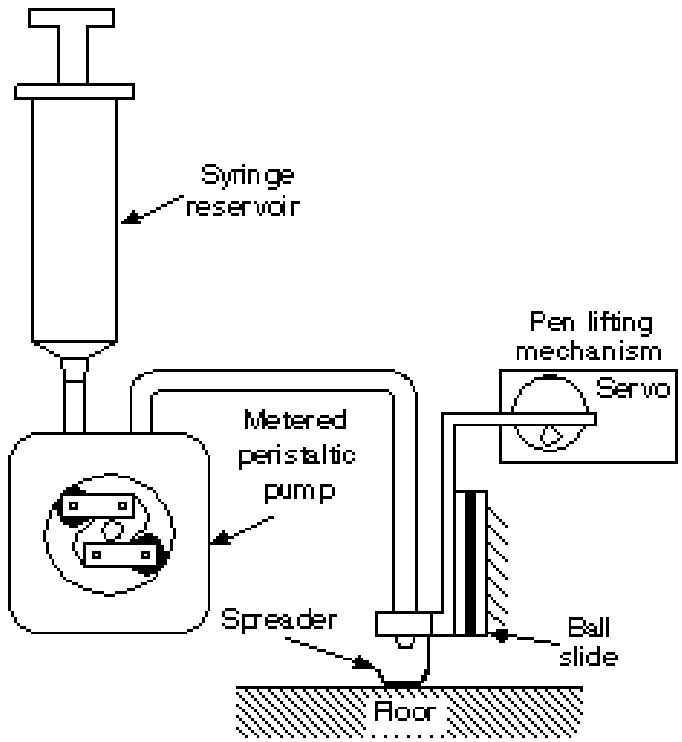


Recommended book for further reading:
Odour Sensing for Mobile Robots, Russell, World Scientific

Second Approach

- Andrew Rusell's RAT robot lays a camphor trail and then follows it back to its starting point.

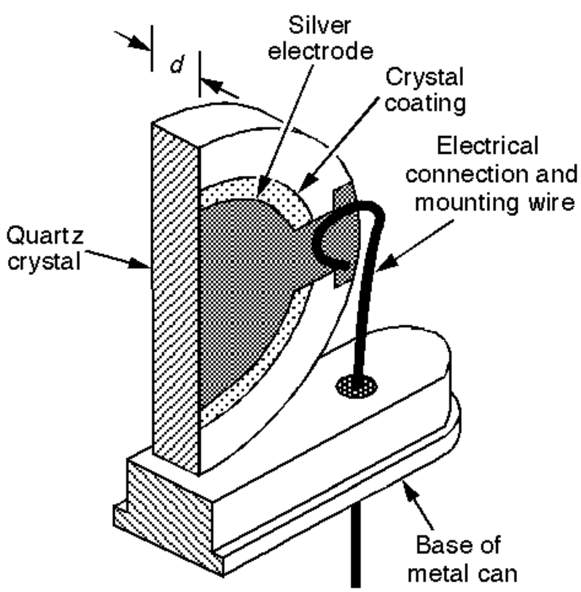


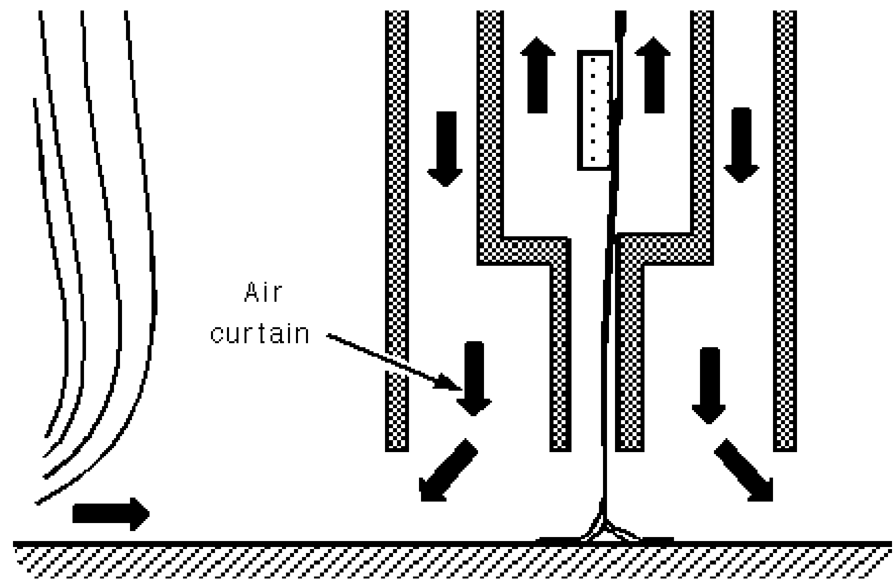


Recommended book for further reading:
Odour Sensing for Mobile Robots, Russell, World Scientific

Second Approach

- Andrew Rusell’s RAT robot lays a camphor trail and then follows it back to its starting point.

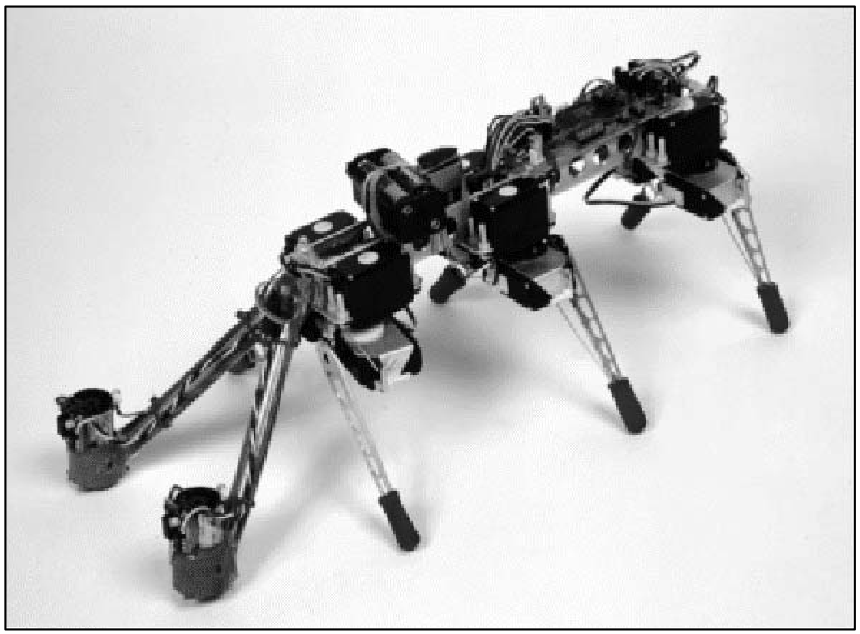




Recommended book for further reading:
Odour Sensing for Mobile Robots, Russell, World Scientific

Second Approach

- Andrew Rusell’s hexapod robot follows a camphor trail.



Recommended book for further reading:
Odour Sensing for Mobile Robots, Russell, World Scientific

Second Approach

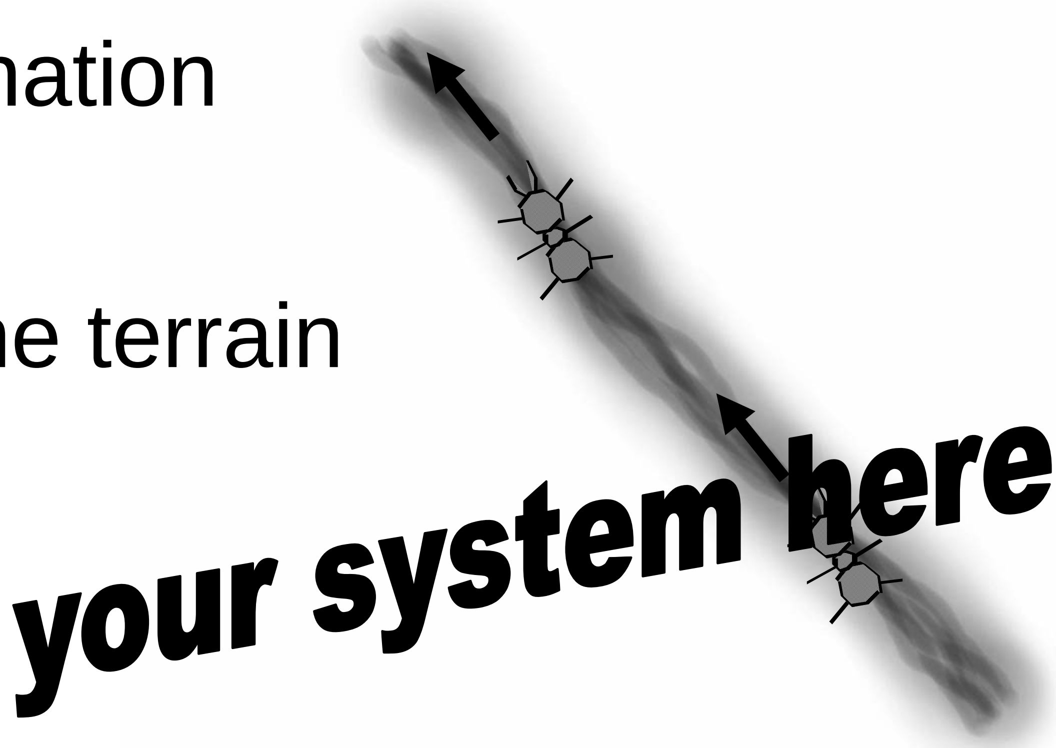
- Andrew Rusell's hexapod robot follows a camphor trail.

Recommended book for further reading:
Odour Sensing for Mobile Robots, Russell, World Scientific

Second Approach

- No location estimates
- No planning
- No explicit coordination

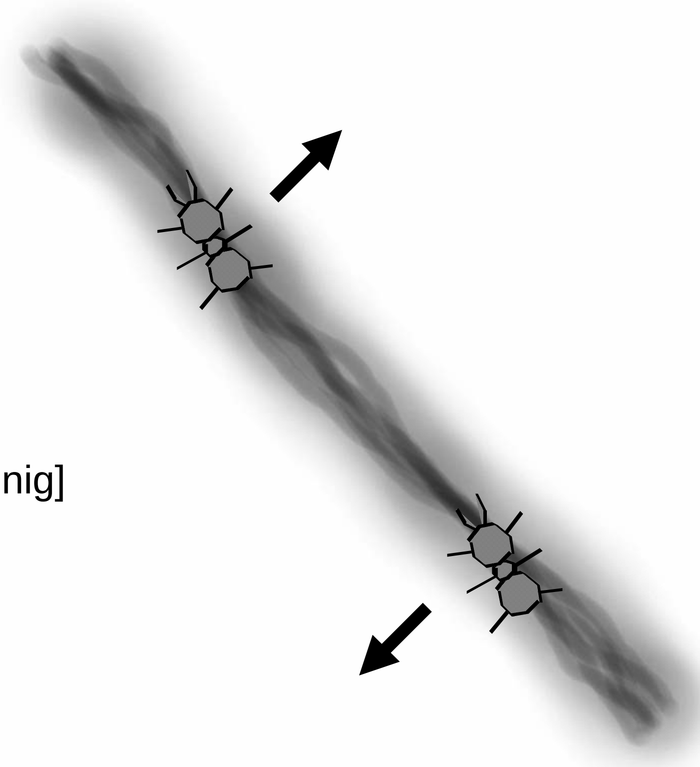
- Leaving trails in the terrain
  - Short-lived trails
    - Heat [Russell]
    - Odor [Russell et al.]
    - Alcohol [Sharpe et al.]



Second Approach

- No location estimates
- No planning
- No explicit coordination

- Leaving trails in the terrain
  - Long-lived trails [Svennebring and Koenig]



Structure

- Motivation
- Real-time search
  - Analytical evaluation
  - Experimental evaluation
- Results on real-time search
- Application to ant robots and results
- Serious application: smart markers

Real-Time Search

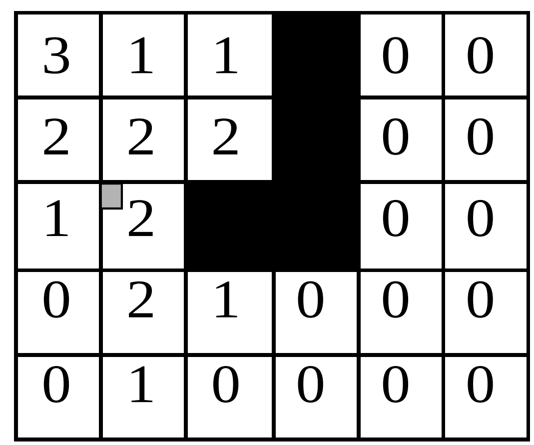
- Real-time search methods provide an interesting means for coordinating single ant robots or teams of ant robots that cover known or unknown terrain once or repeatedly.
- They leave markings in the terrain, similar to what some ants do.
- The ant robots robustly cover terrain even if the robots are moved without realizing this, some robots fail, and some markings get destroyed. The robots do not even need to be localized.

Node Counting

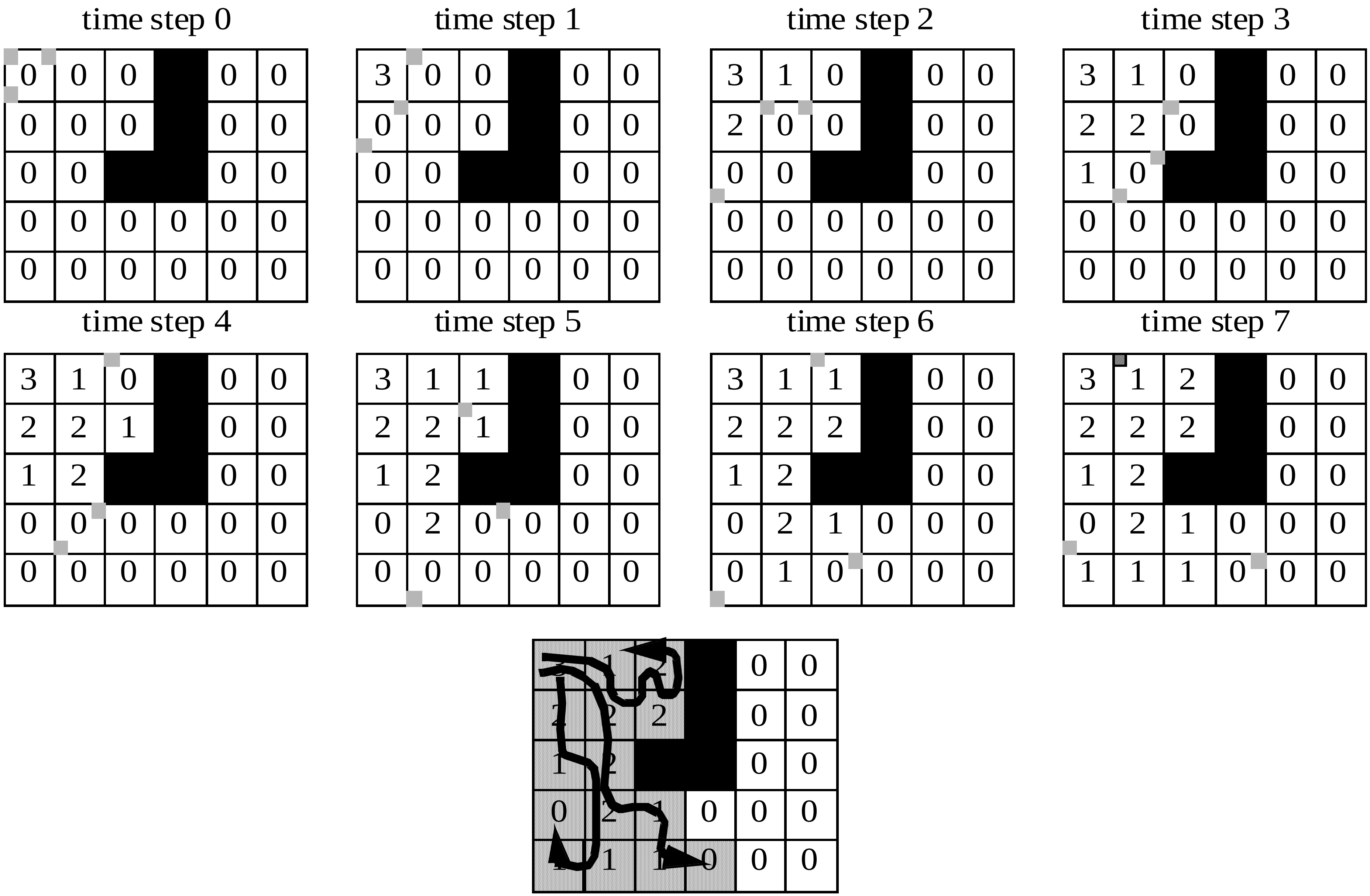
Initially, the u-values u(s) are zero for all cells s.

1. s := start cell
2. s’ := a cell adjacent to cell s with a minimal u-value
3. u(s) := 1 + u(s)
4. move the ant robot to cell s’
5. go to 2

What do the u-values mean?
Where would you move?



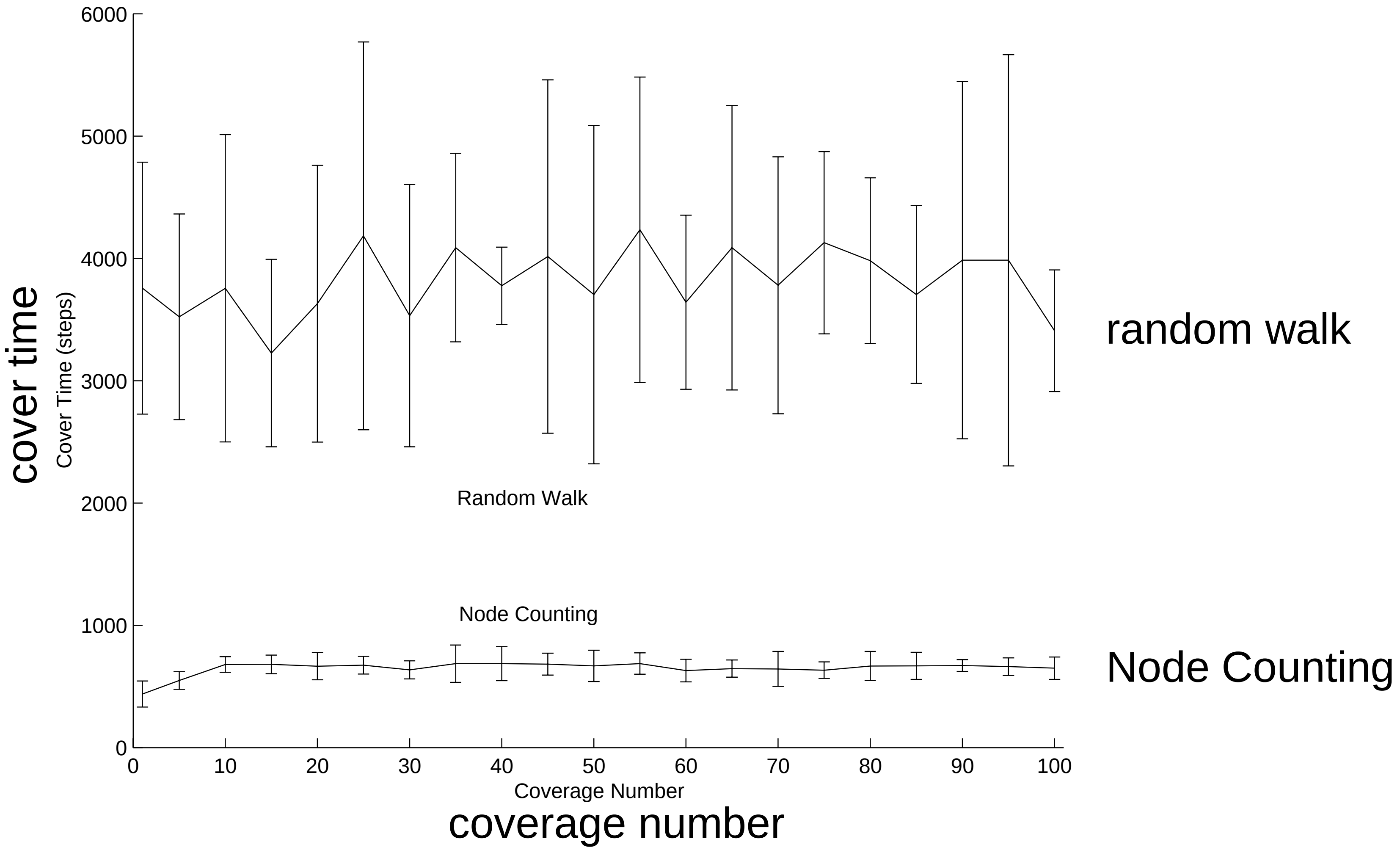
Coverage: Node Counting



Coverage: Node Counting

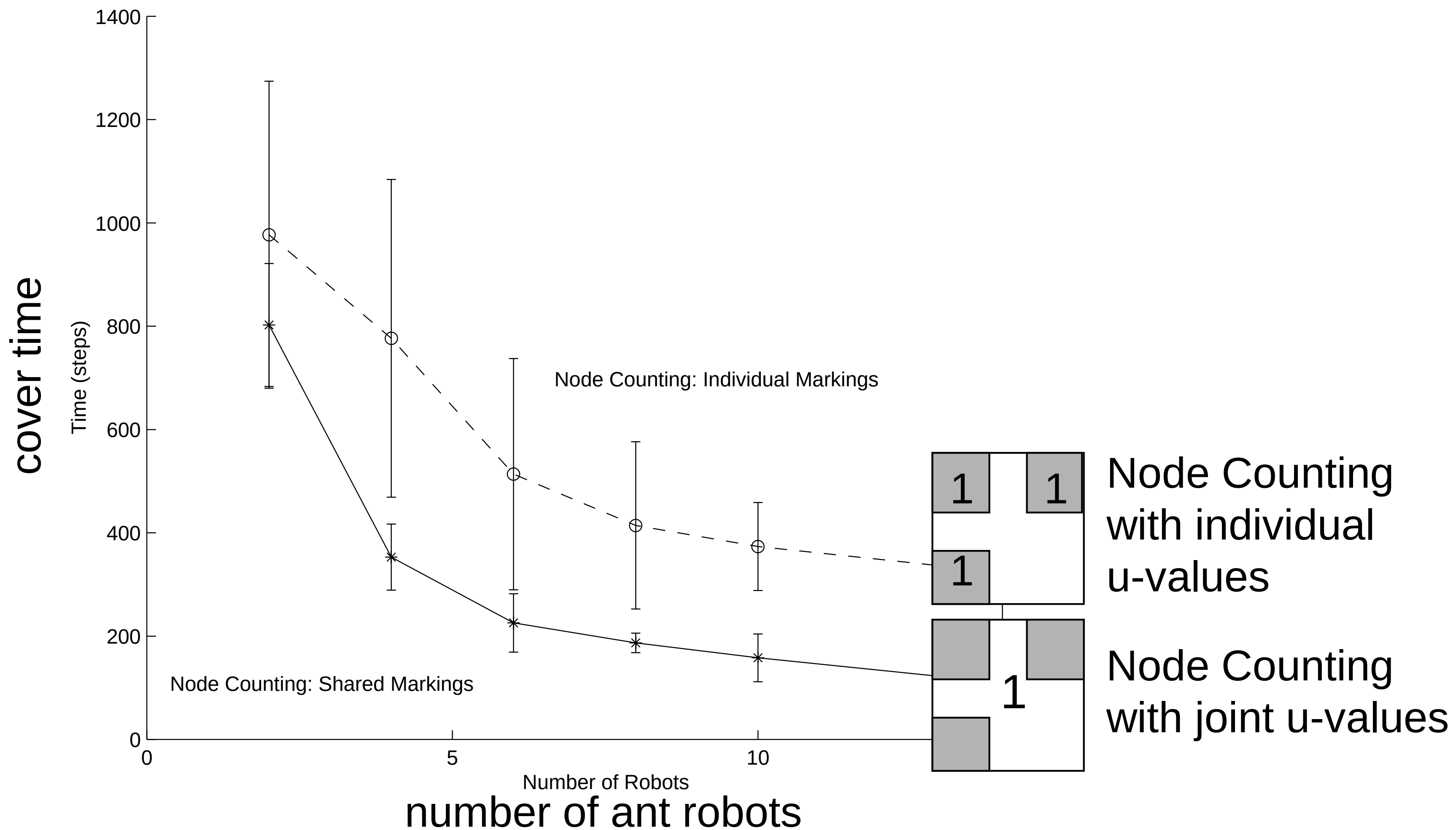
Coverage: Node Counting

- The u-values coordinate ant robots.



Coverage: Node Counting

- Sharing the u-values coordinates ant robots.



Real-Time Search Methods

Initially, the u-values u(s) are zero for all cells s.

1. s := start cell
2. s' := a cell adjacent to cell s with a minimal u-value
3. u(s) := 1 + u(s) — Node Counting
   or u(s) := 1 + u(s') — Korf's LRTA*
   or if u(s) ≤ u(s') then u(s) := 1 + u(s) — Wagner's Rule
   or u(s) := max(1+u(s), 1+u(s')) — Thrun's Rule
4. move the ant robot to cell s'
5. go to 2

your method here

What do the u-values mean?
Where would you move?

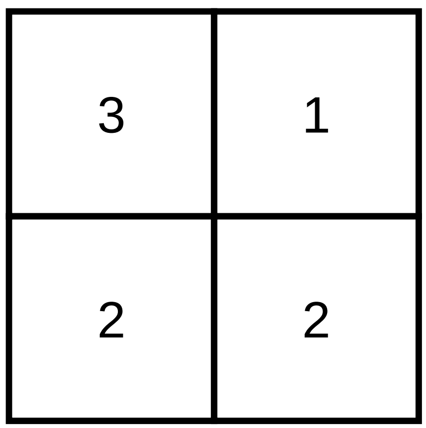
3	1	1		0	0
2	2	2		0	0
1	2			0	0
0	2	1	0	0	0
0	1	0	0	0	0

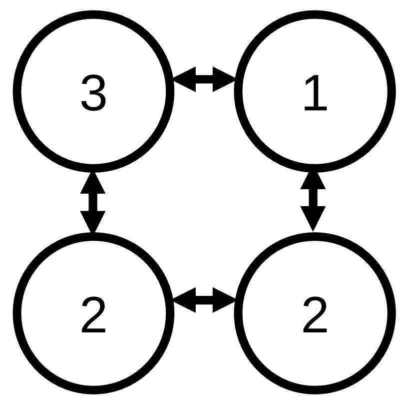
Structure

- Motivation
- Real-time search
  - Analytical evaluation
  - Experimental evaluation
- Results on real-time search
- Application to ant robots and results
- Serious application: smart markers

Real-Time Search Methods

- From grids to directed graphs



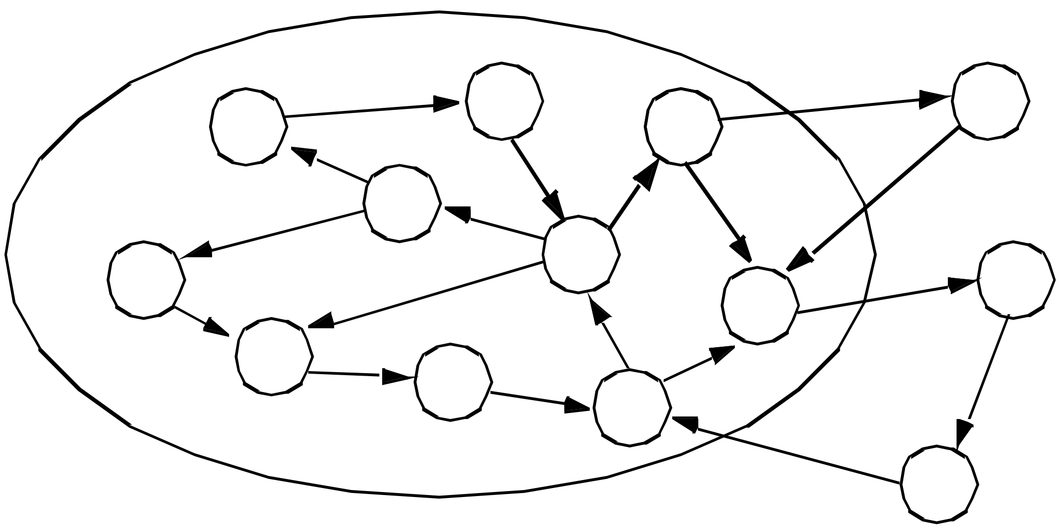


Real-Time Search Methods

Theorem:

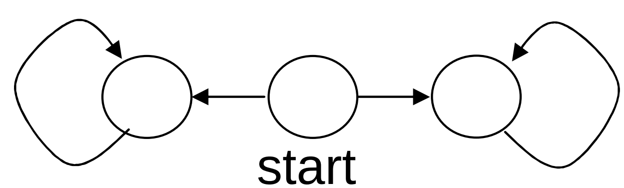
Teams of ant robots that all use the same real-time search method cover all strongly connected graphs repeatedly.

Proof:



QED

The graphs need to be strongly connected:

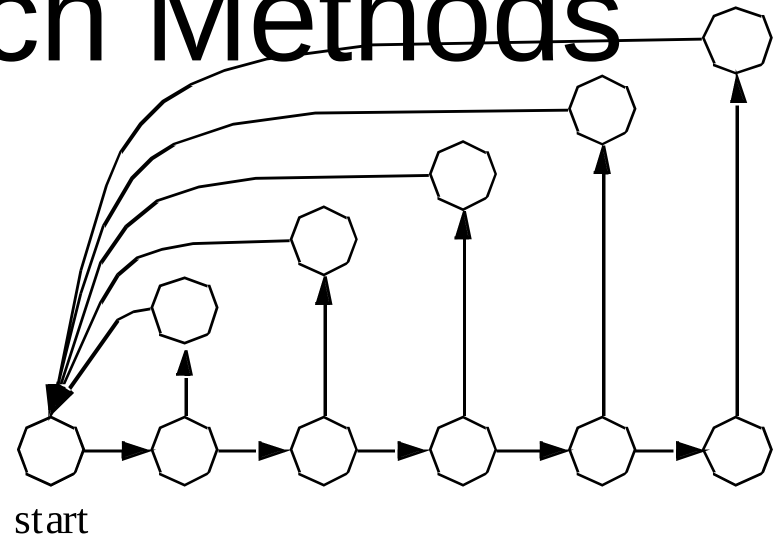


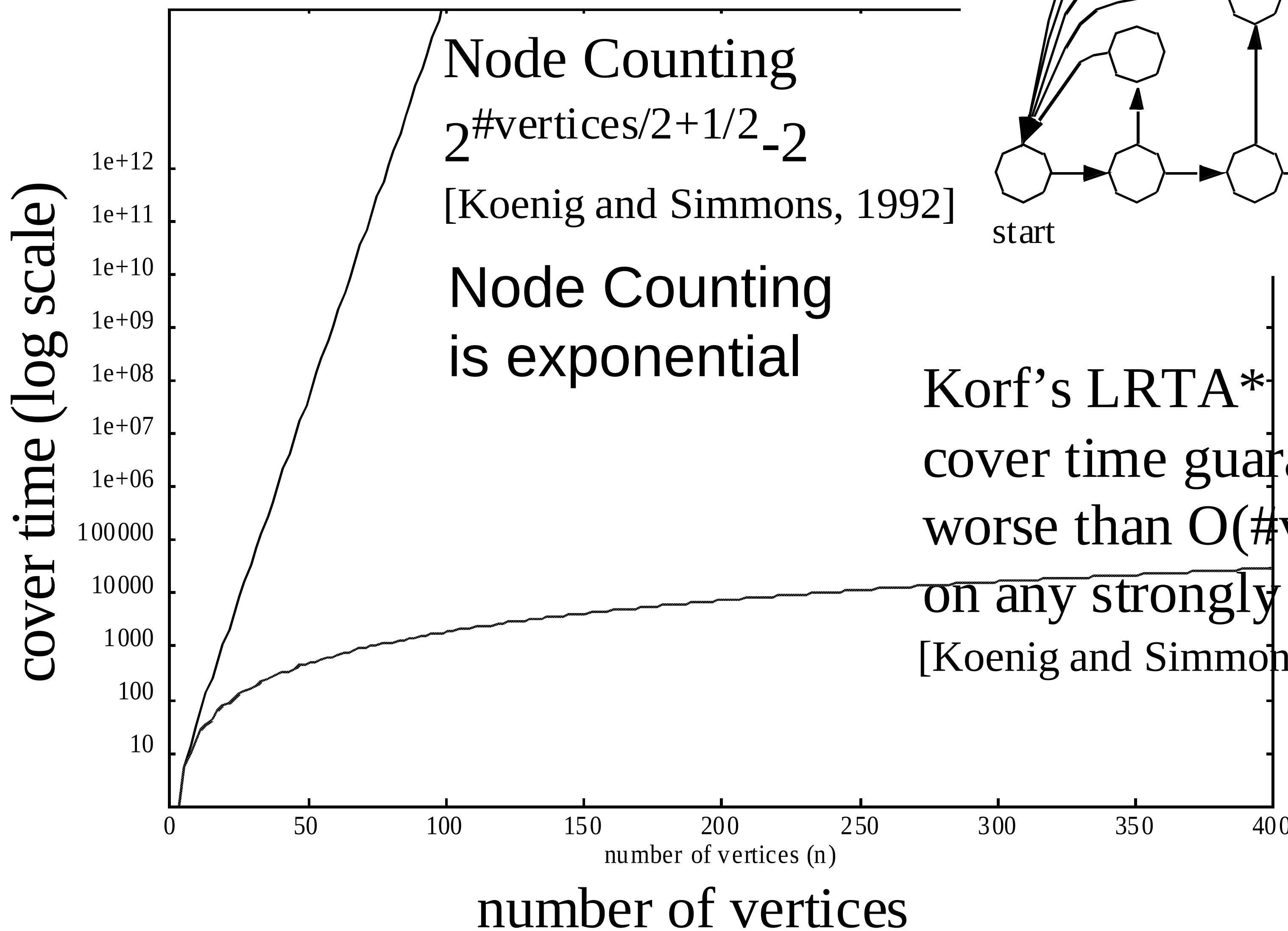
Real-Time Search Methods

- How fast is the coverage (= cover time) in the worst case, that is, if an adversary can choose the graph topology, the start vertex and the tie-breaking rule?
  - Node Counting: exponential
  - Korf's LRTA*: polynomial
  - Wagner's Rule: polynomial
  - Thrun's Rule: polynomial

Real-Time Search Methods

- strongly connected directed graphs



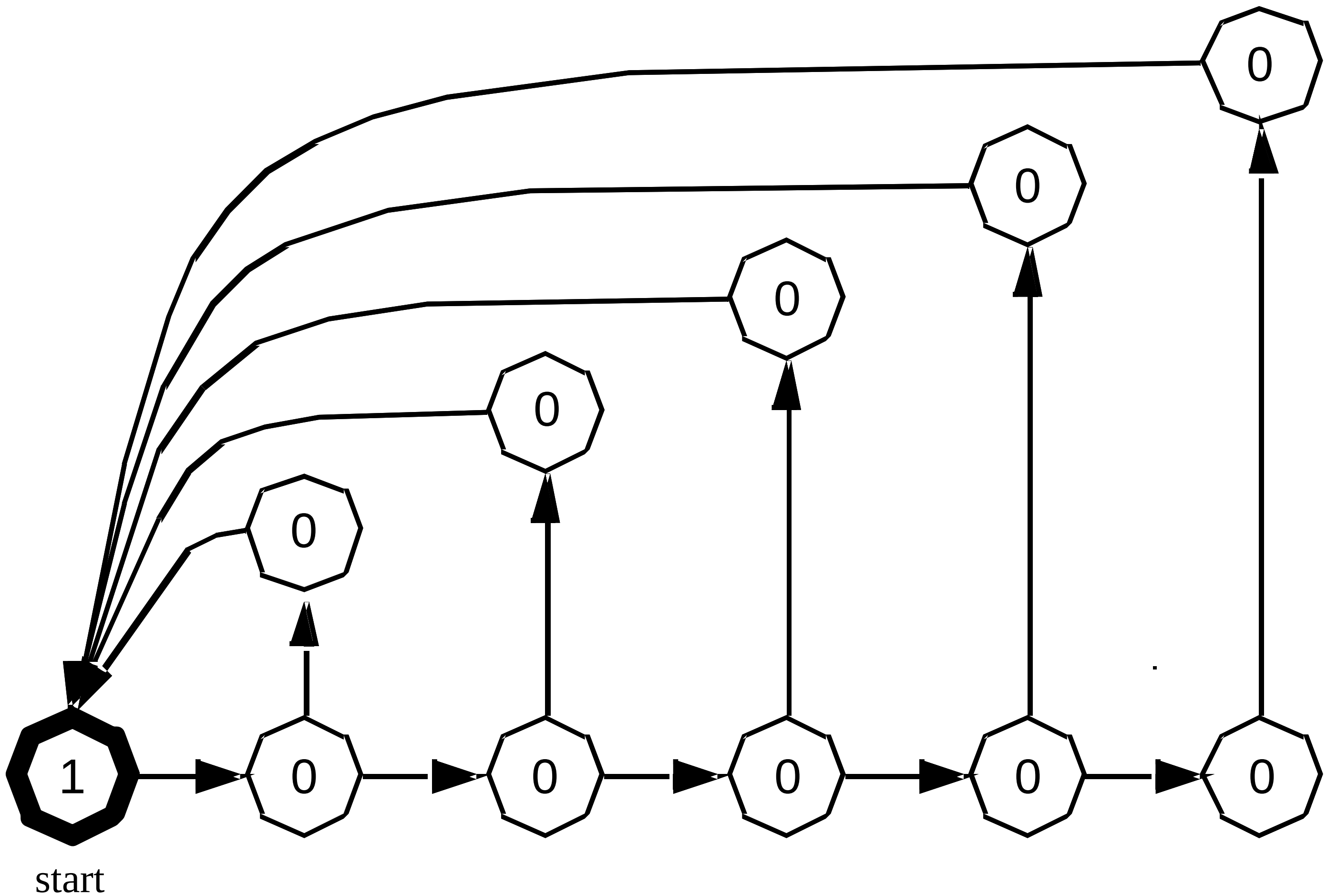


Node Counting
$2^{\#vertices/2+1/2}-2$
[Koenig and Simmons, 1992]

Node Counting
is exponential

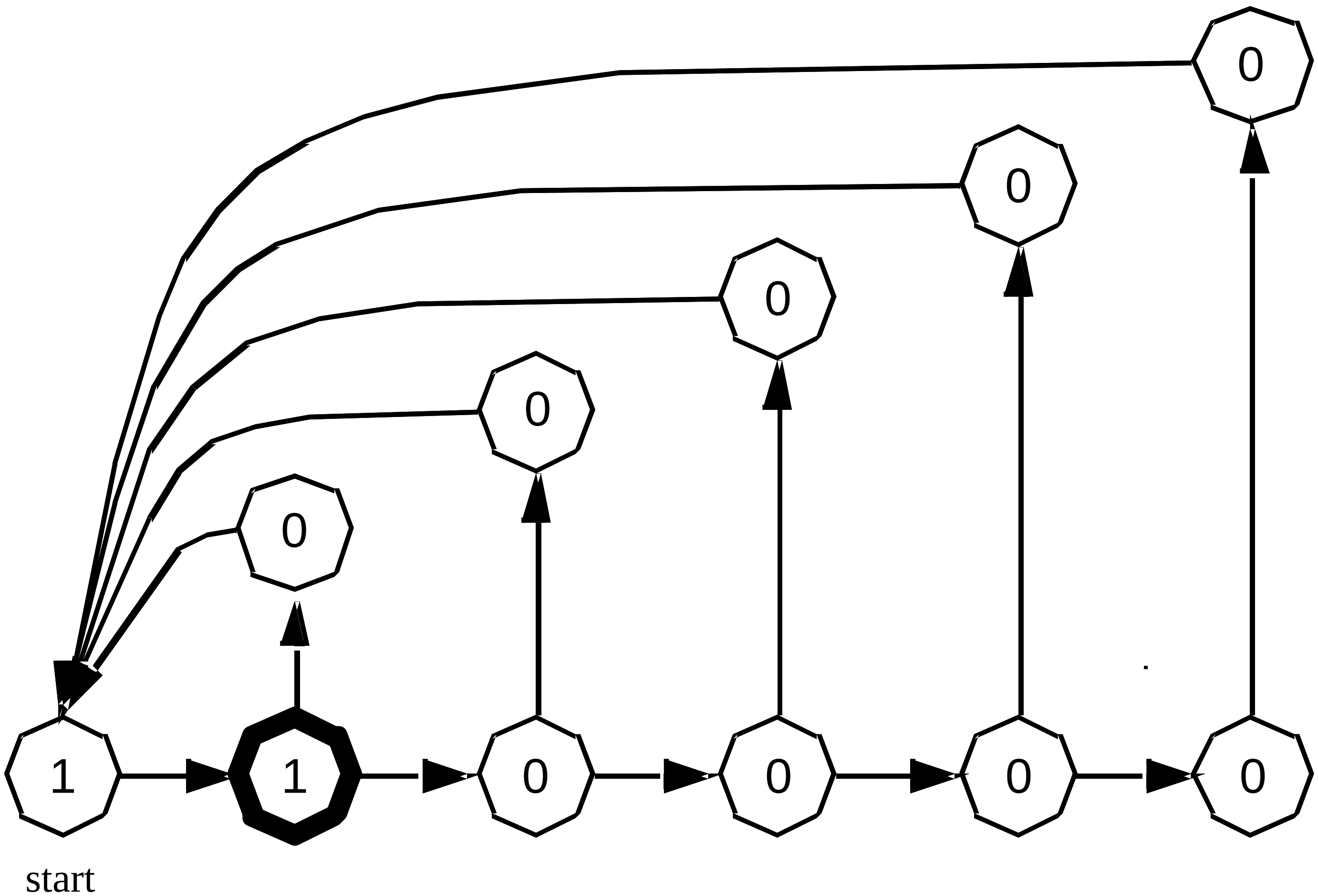
Korf's LRTA*
cover time guaranteed to be no worse than O(#vertices diameter) on any strongly connected graph
[Koenig and Simmons, 1992]

Real-Time Search Methods

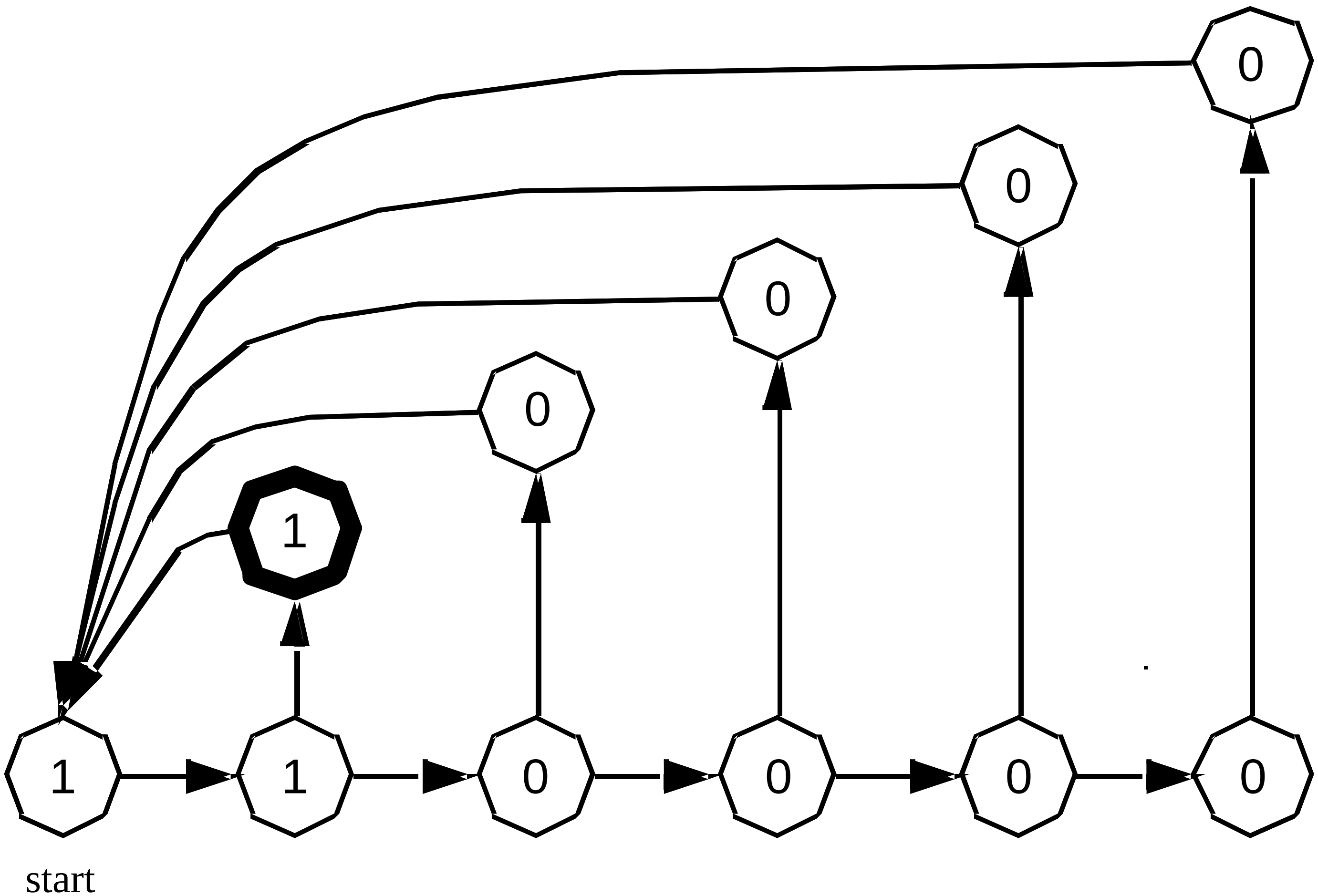


we break ties by going “up”

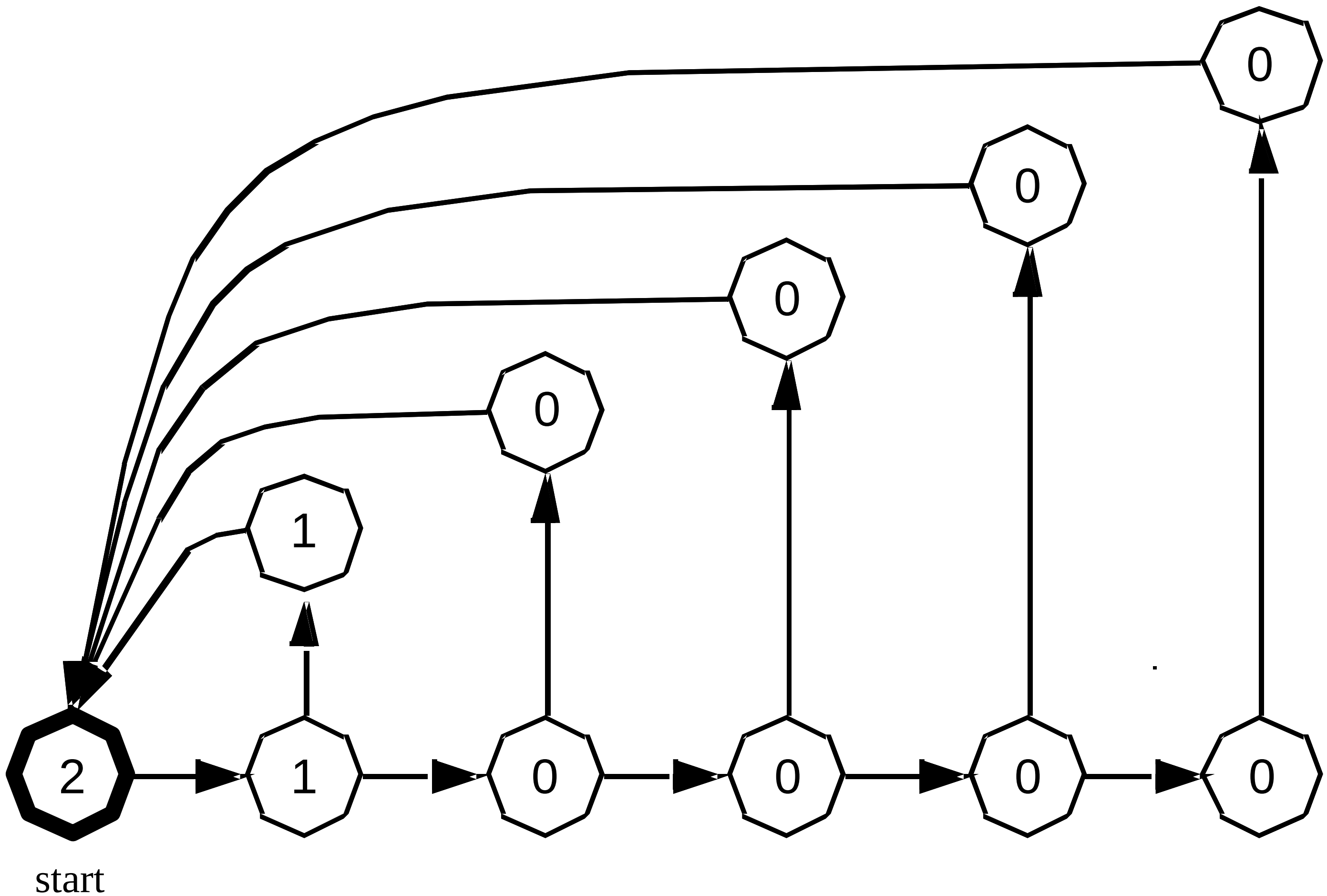
Real-Time Search Methods



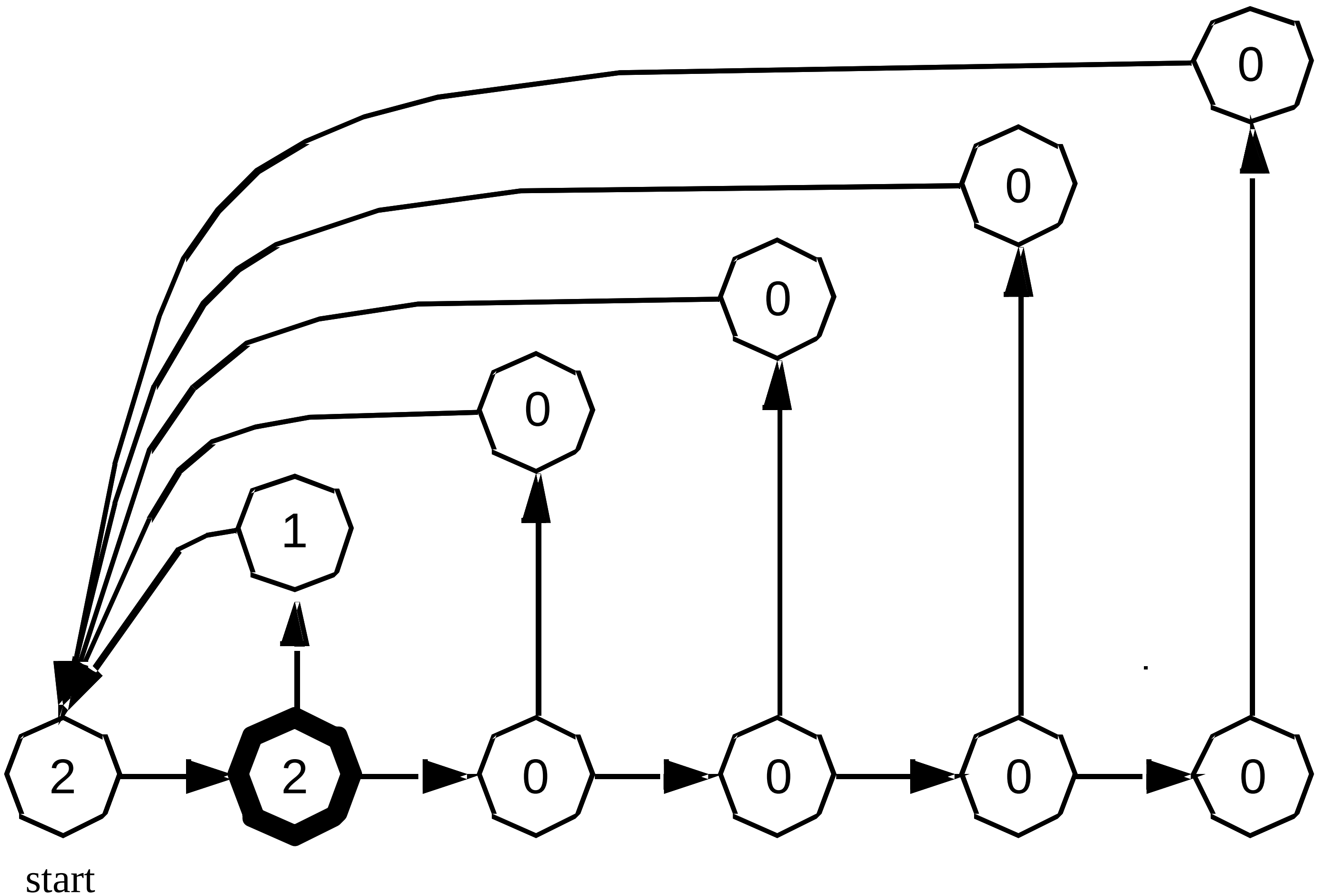
Real-Time Search Methods



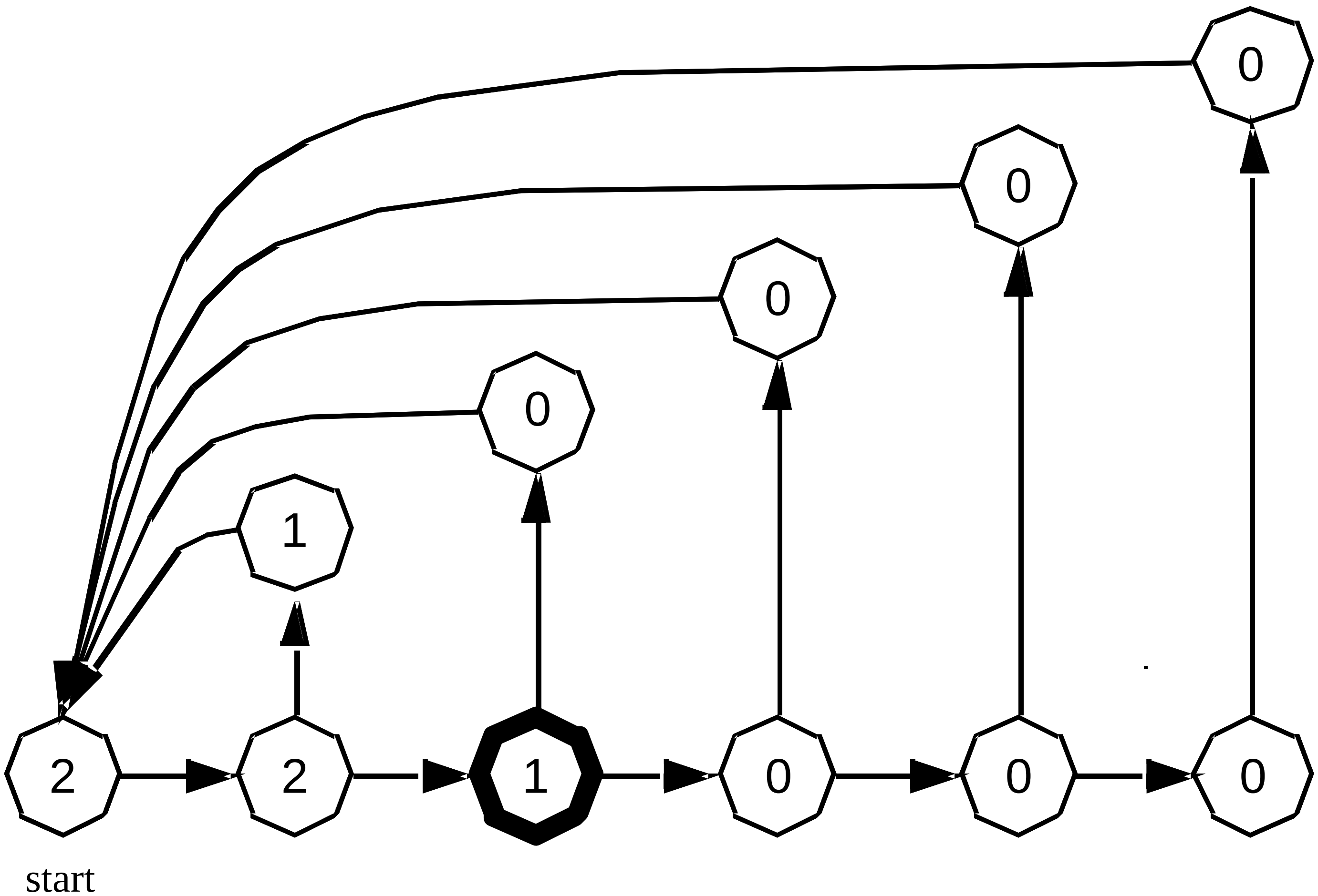
Real-Time Search Methods



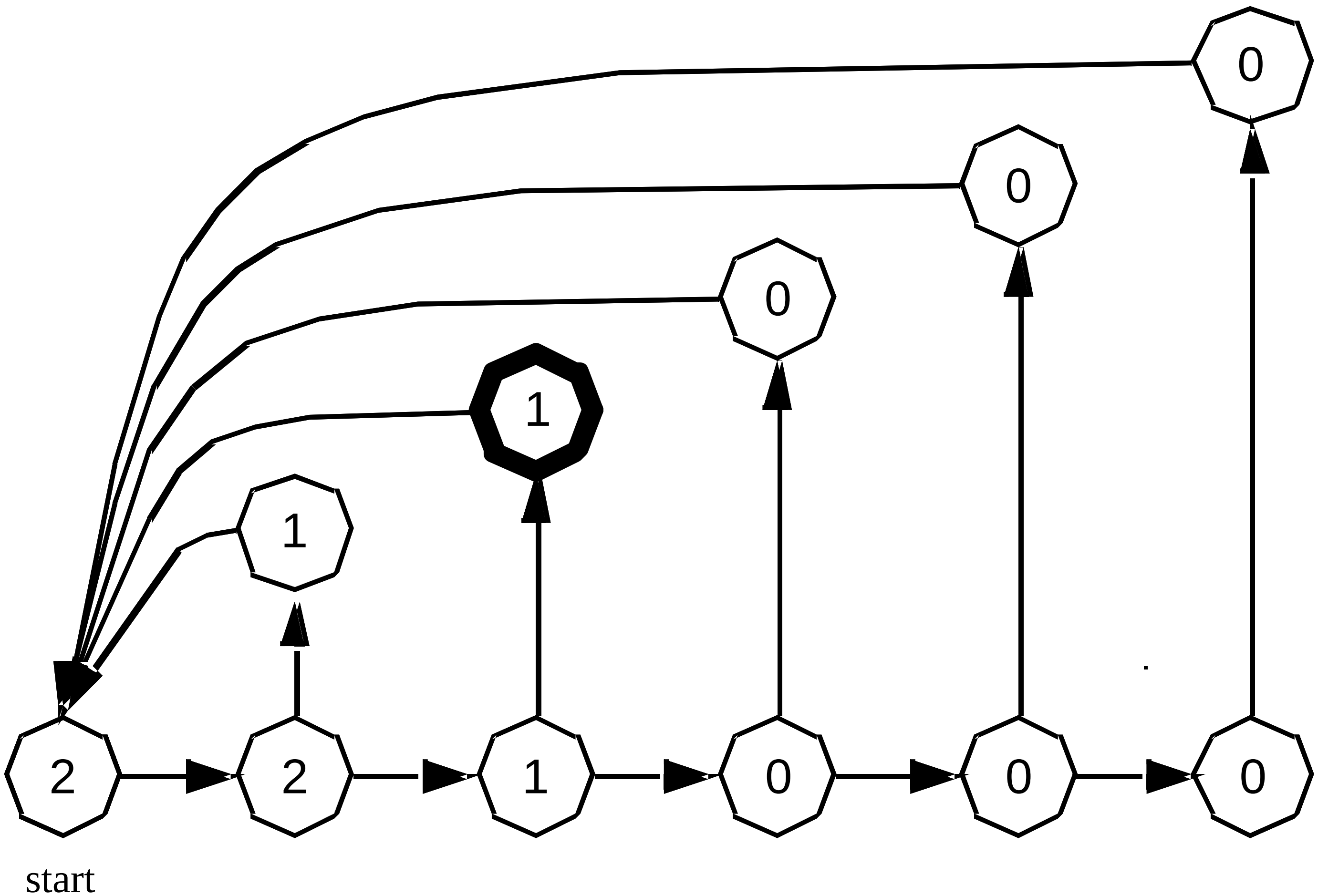
Real-Time Search Methods



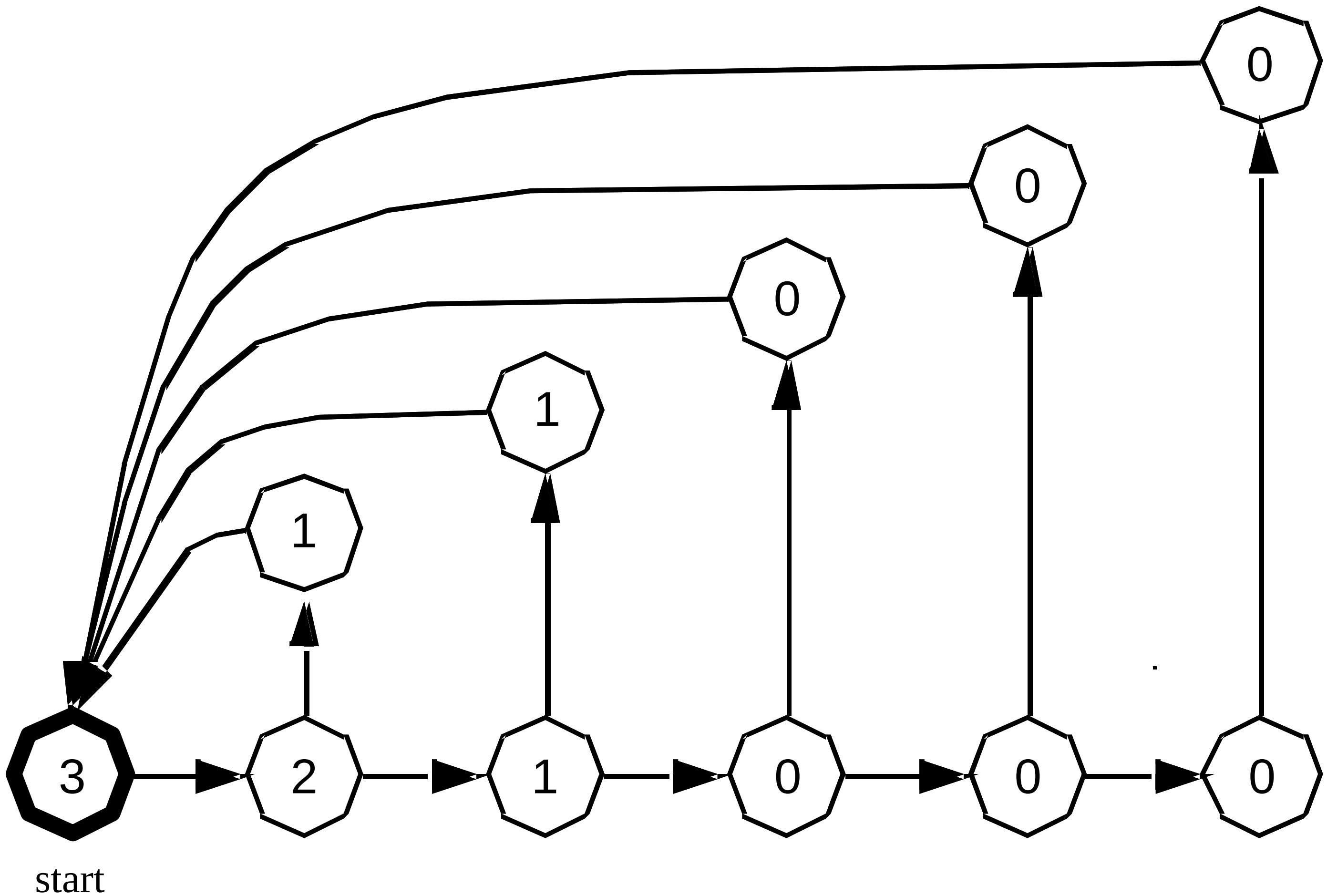
Real-Time Search Methods



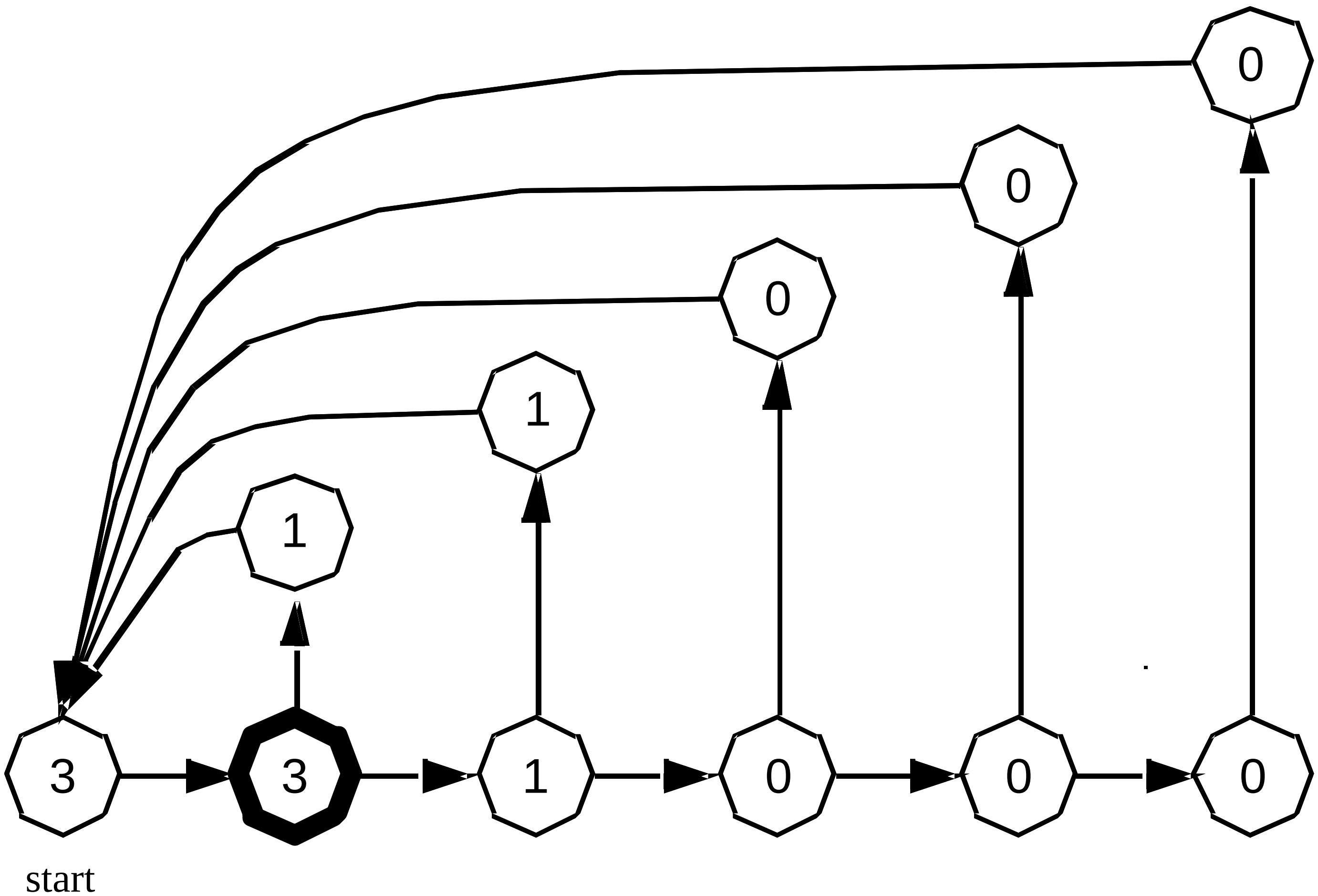
Real-Time Search Methods



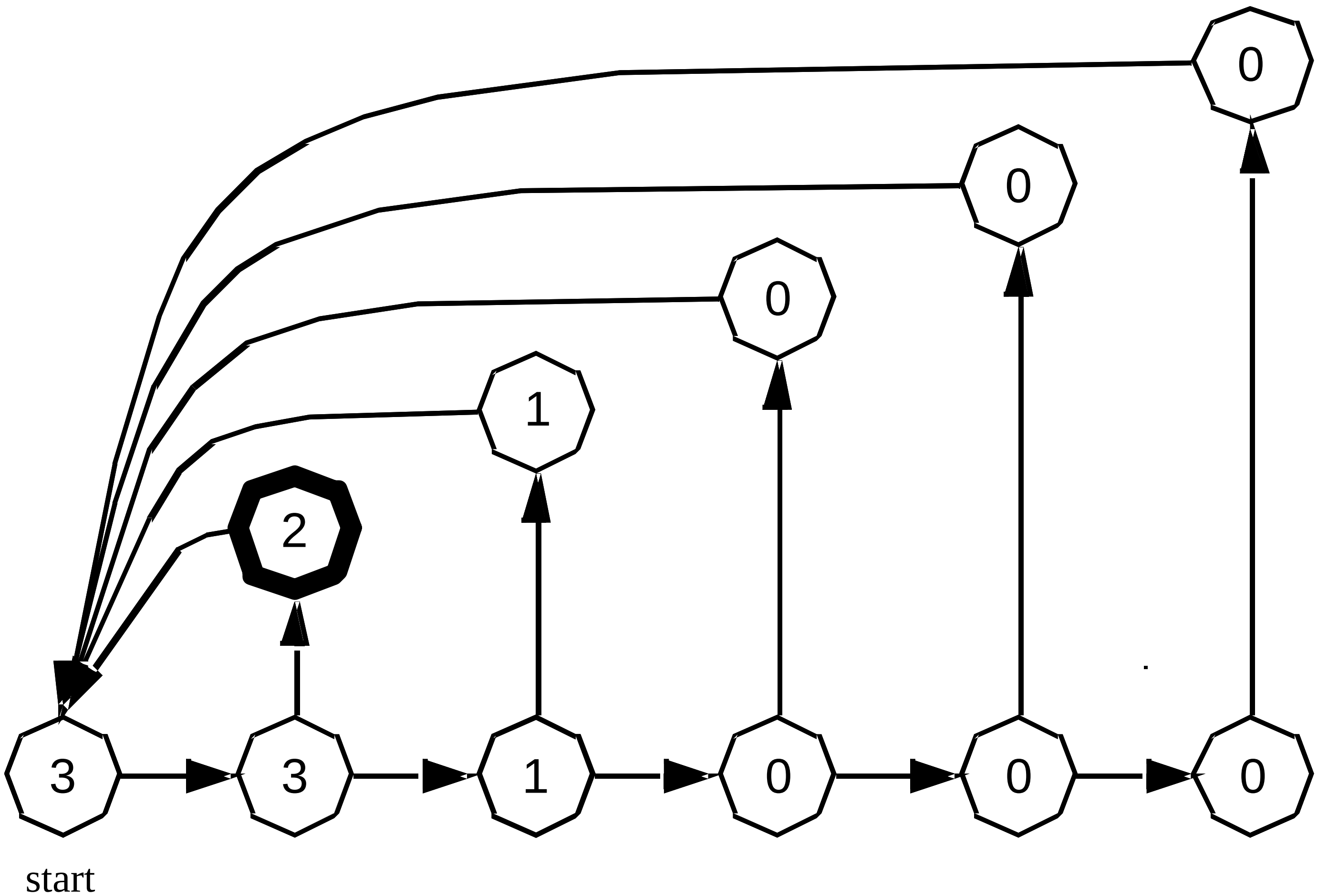
Real-Time Search Methods



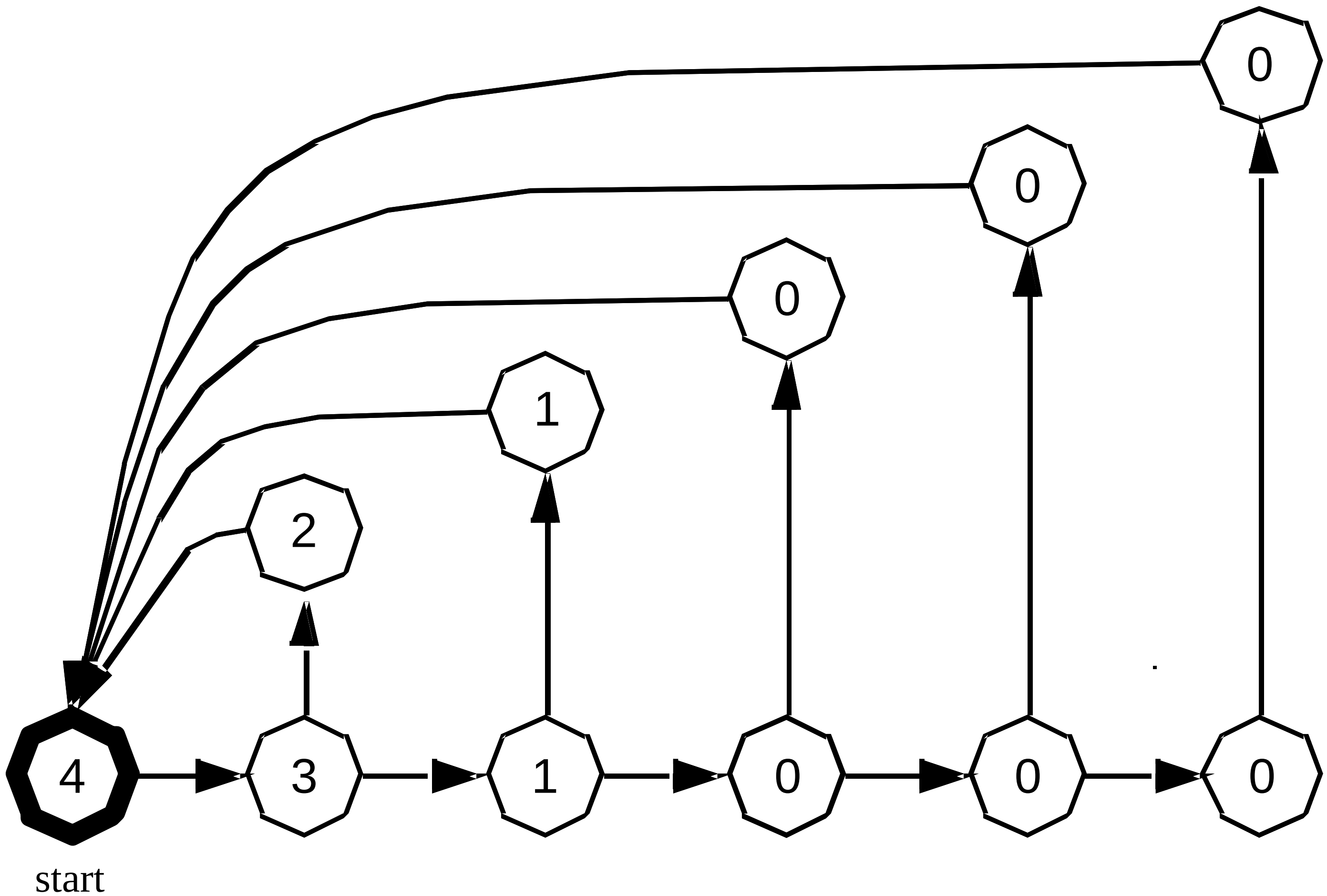
Real-Time Search Methods



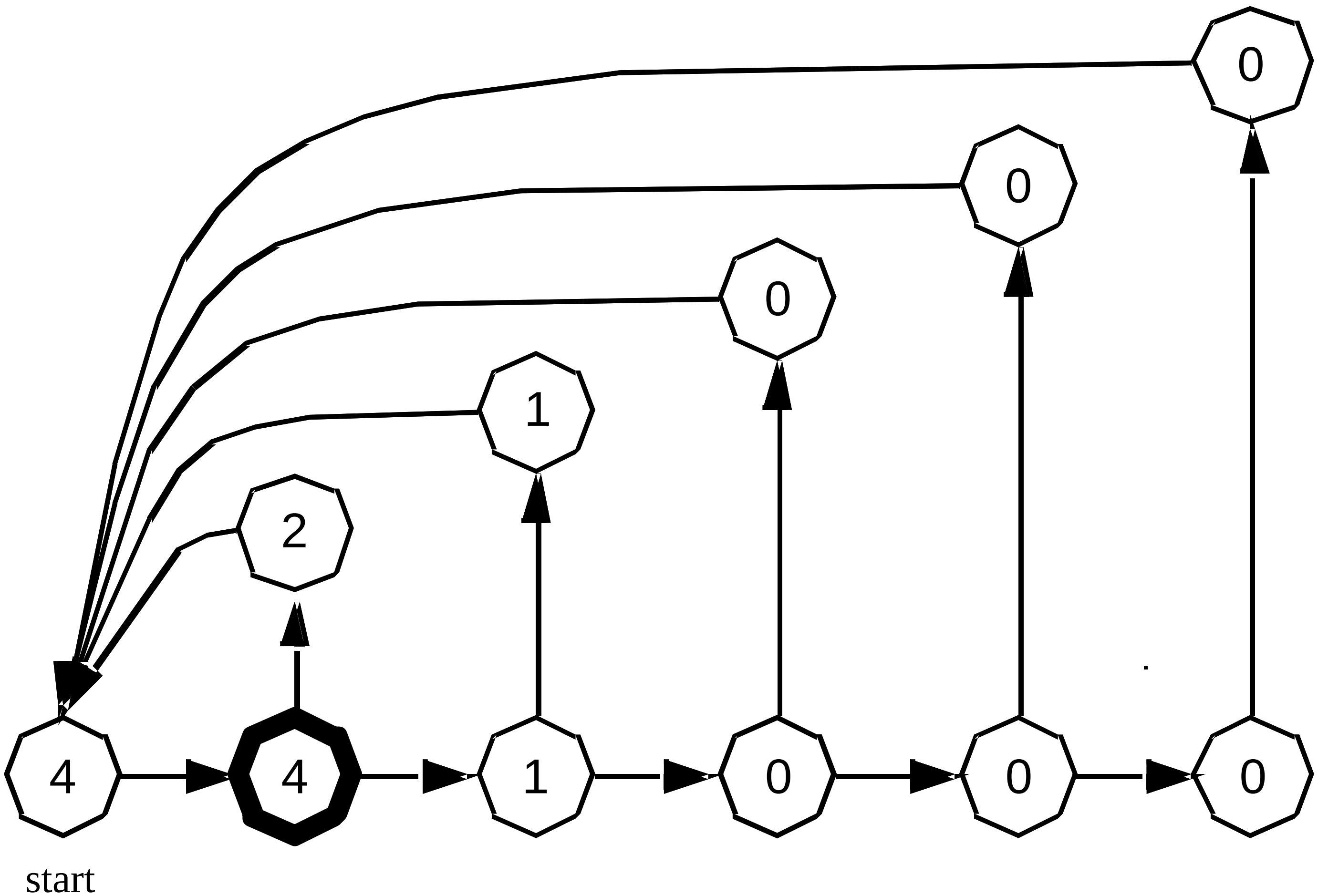
Real-Time Search Methods



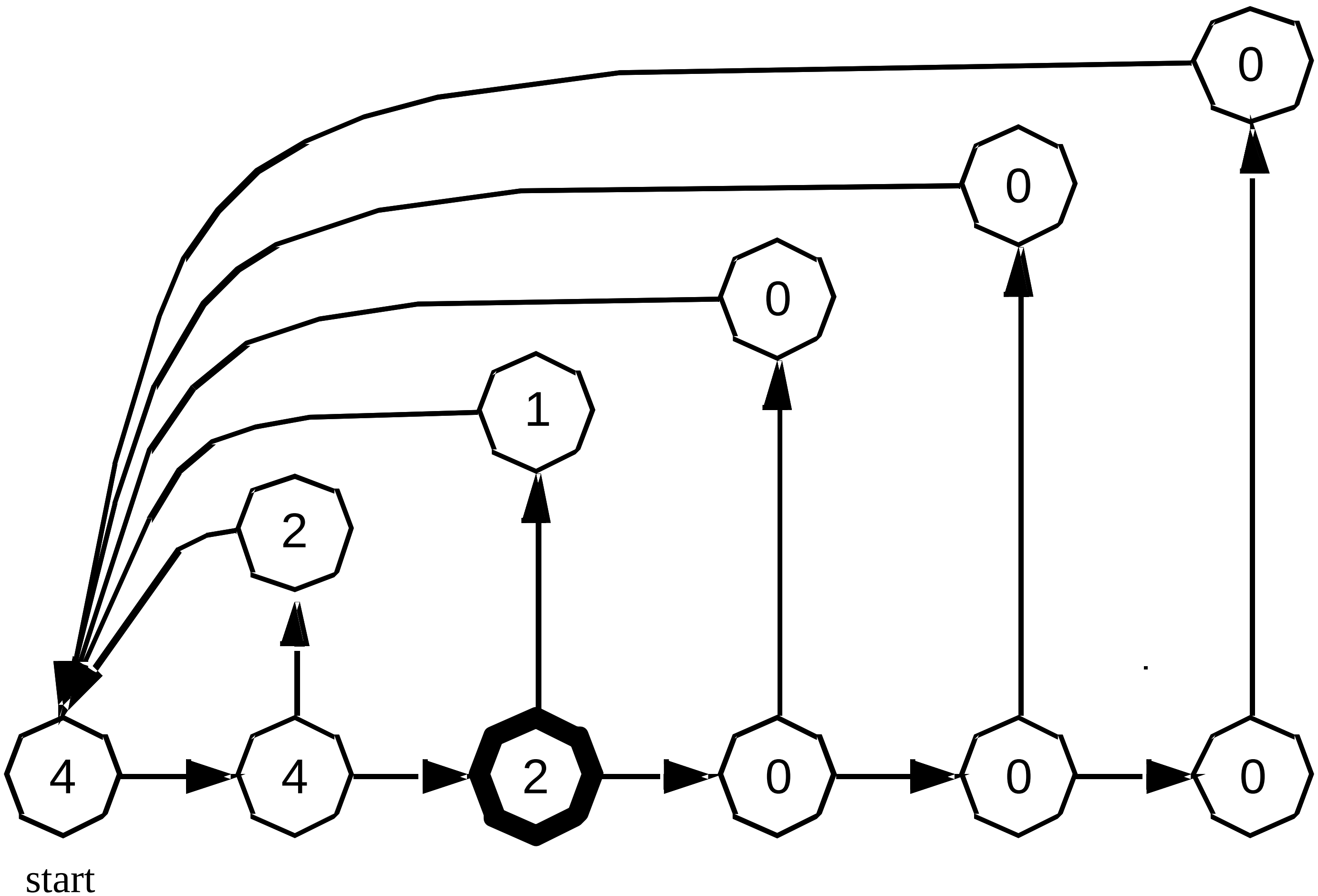
Real-Time Search Methods



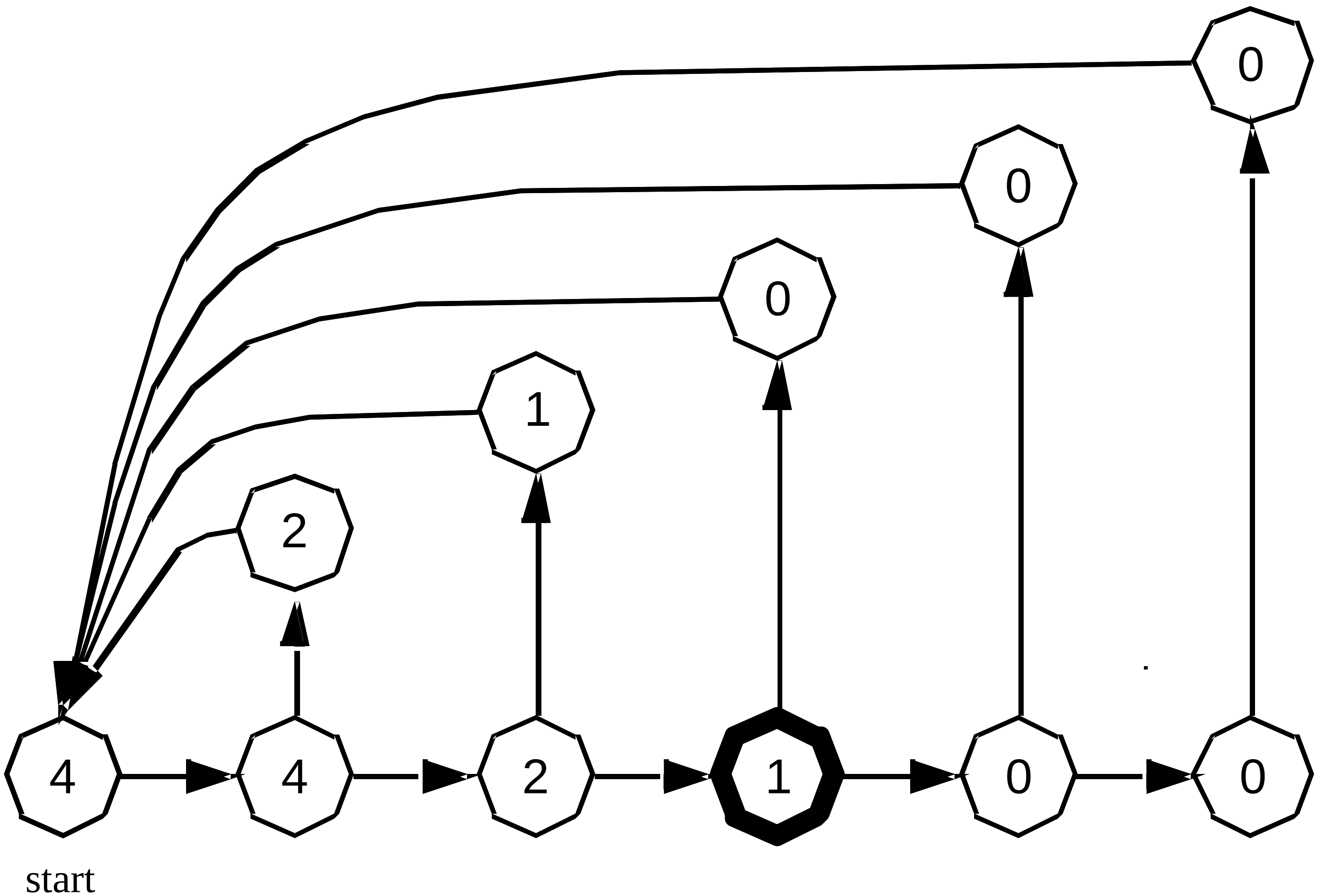
Real-Time Search Methods



Real-Time Search Methods



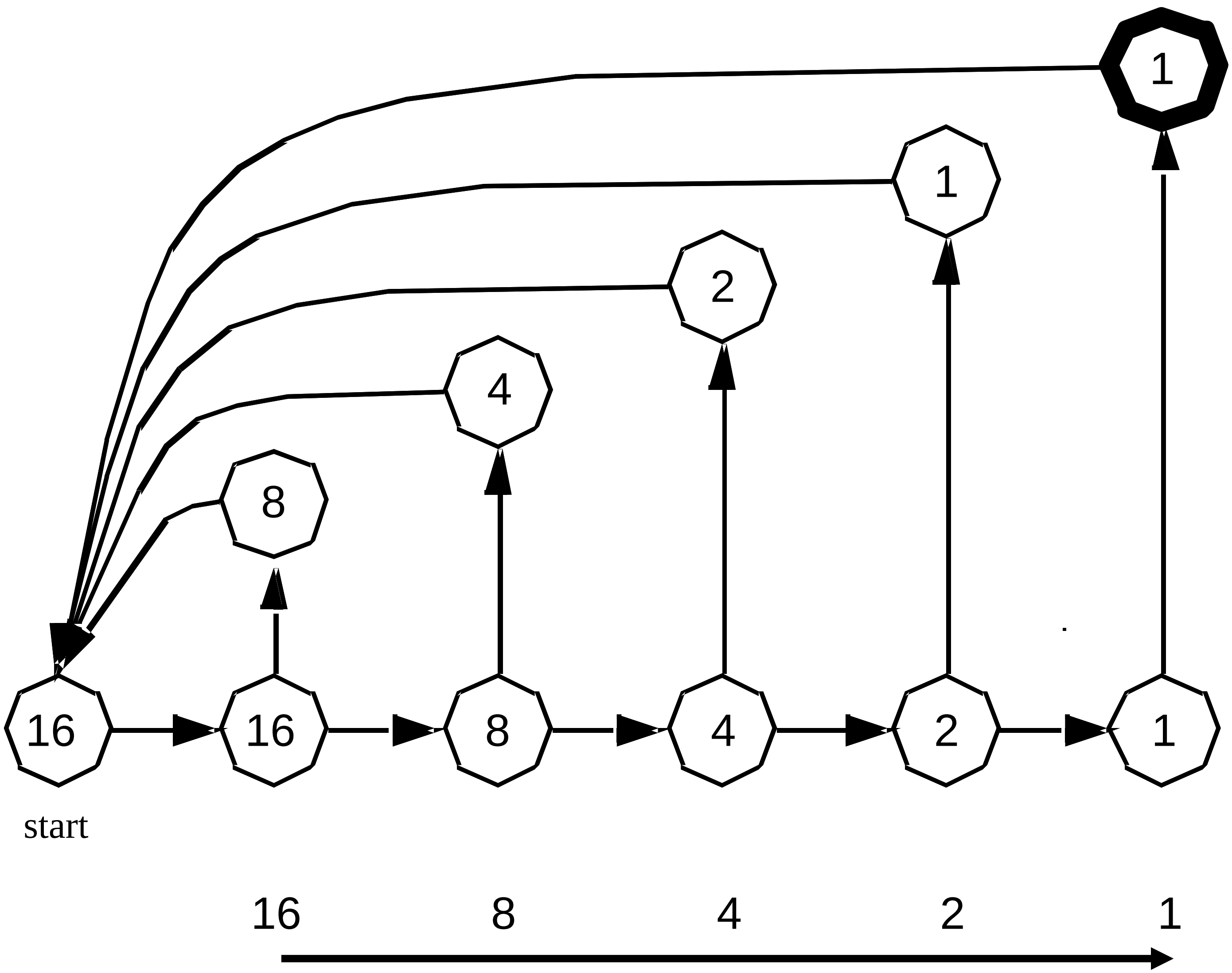
Real-Time Search Methods



Real-Time Search Methods

and so on …

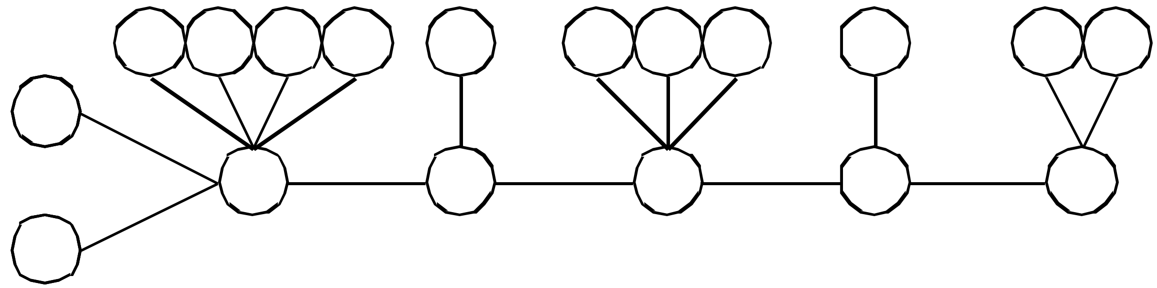
Real-Time Search Methods

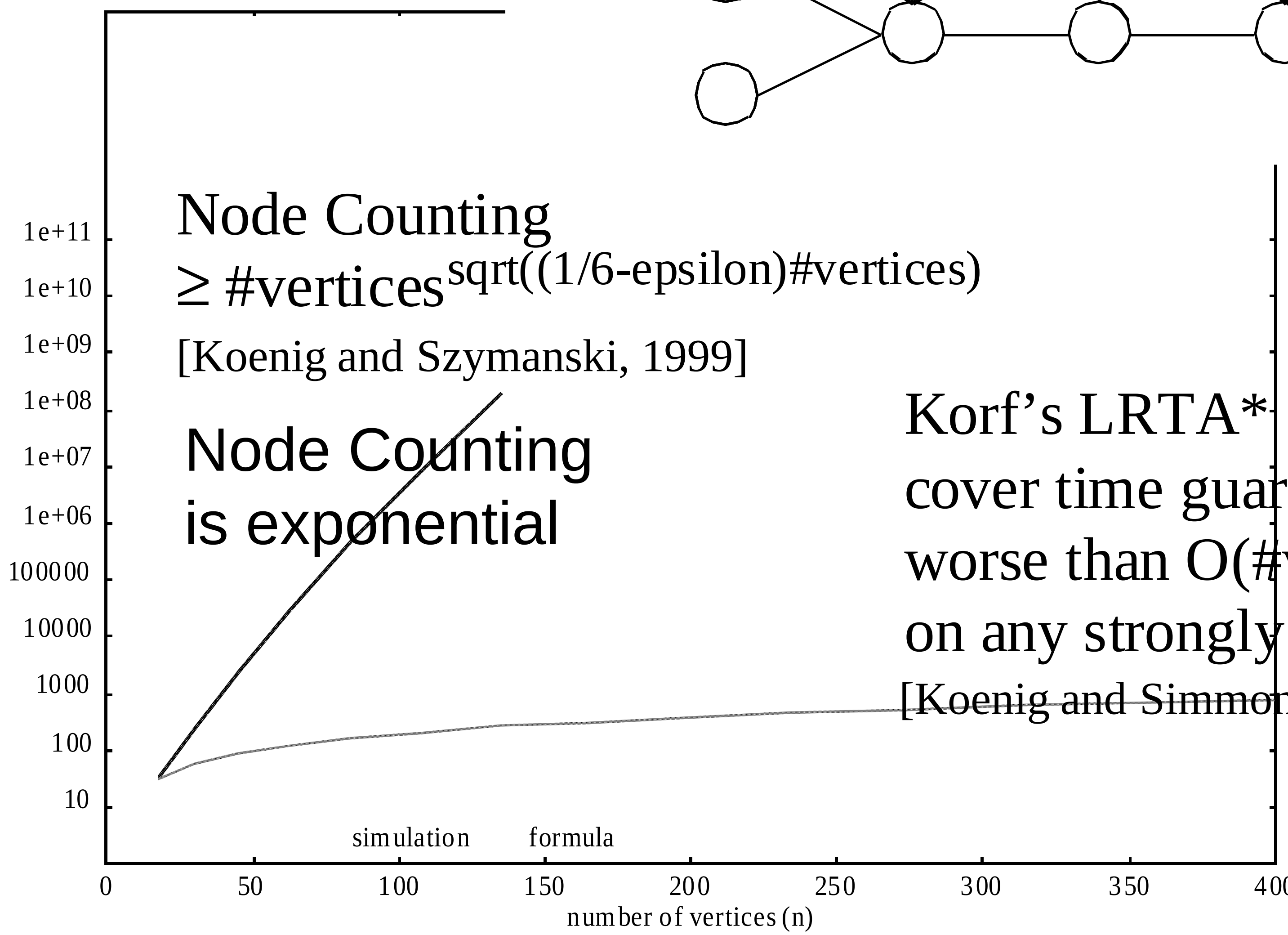


Real-Time Search Methods

- connected undirected graphs

start





cover time (log scale)

Node Counting
$\geq$ #vertices$^{\text{sqrt((1/6-epsilon)\#vertices)}}$
[Koenig and Szymanski, 1999]

Node Counting
is exponential

Korf's LRTA*
cover time guaranteed to be no worse than O(#vertices diameter) on any strongly connected graph
[Koenig and Simmons, 1992]

number of vertices

Real-Time Search Methods

- connected undirected graphs with bounded degree

your proof here

Node Counting

Is is unknown whether Node Counting is polynomial

Korf's LRTA*

cover time guaranteed to be no worse than O(#vertices diameter) on any strongly connected graph

[Koenig and Simmons, 1992]

Real-Time Search Methods

- connected undirected grids

your proof here

Node Counting

Is is unknown whether Node Counting is polynomial

Korf's LRTA*

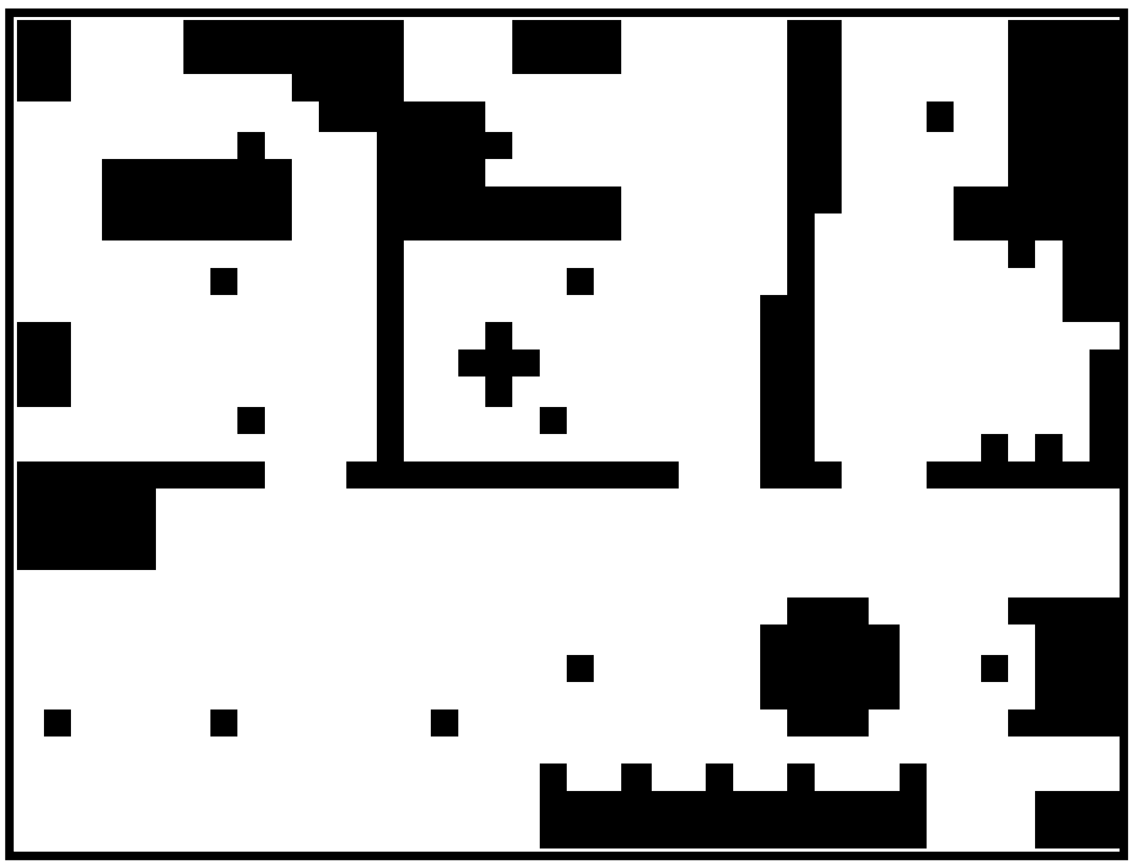
cover time guaranteed to be no worse than O(#vertices diameter) on any strongly connected graph

[Koenig and Simmons, 1992]

Structure

- Motivation
- Real-time search
  - Analytical evaluation
  - Experimental evaluation
- Results on real-time search
- Application to ant robots and results
- Serious application: smart markers

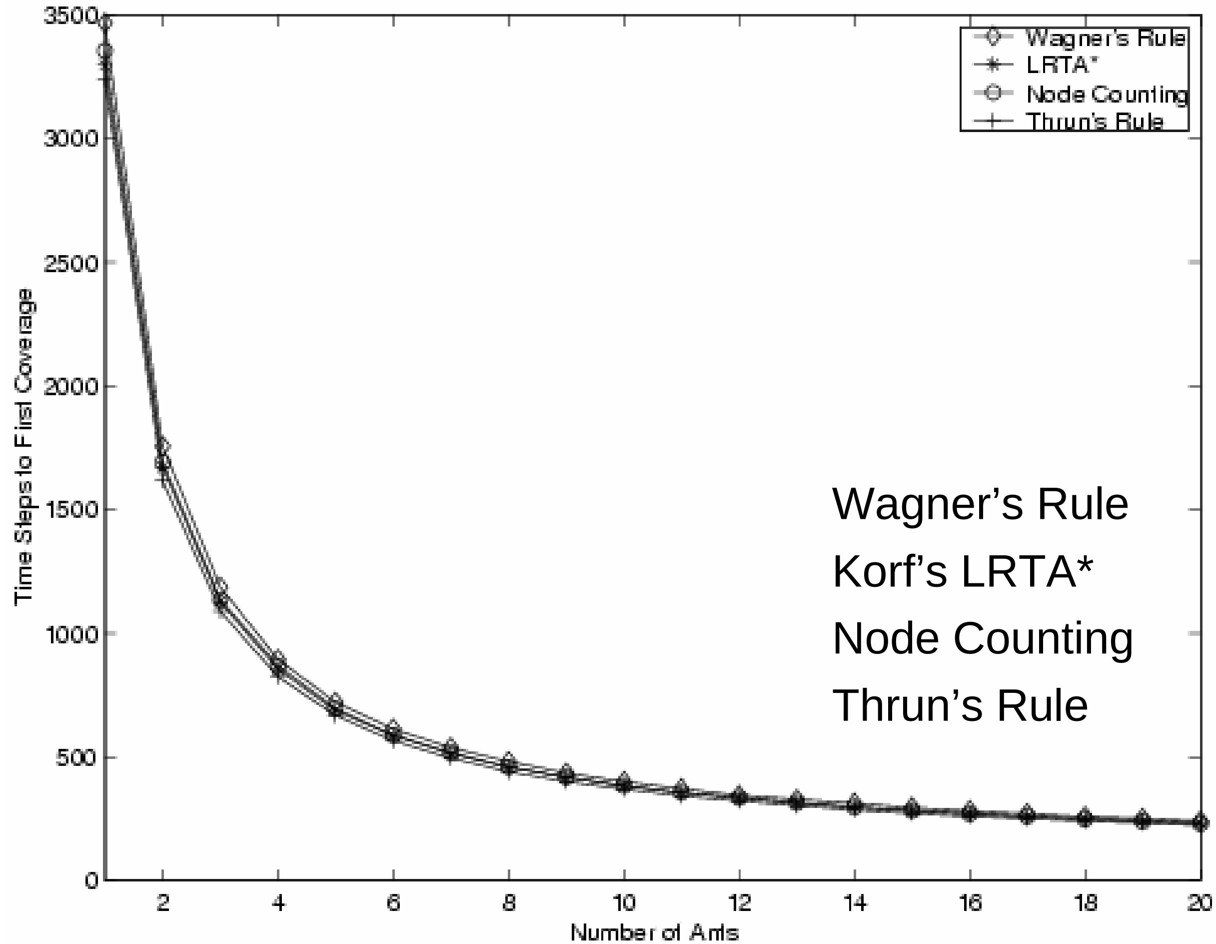
Real-Time Search Methods



Real-Time Search Methods

- How fast is the coverage on average?

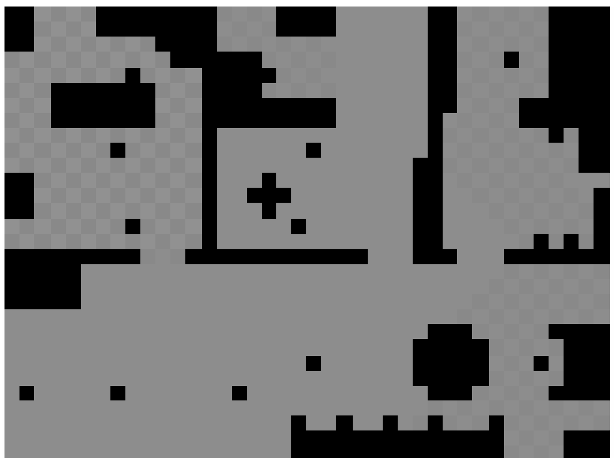
cover time



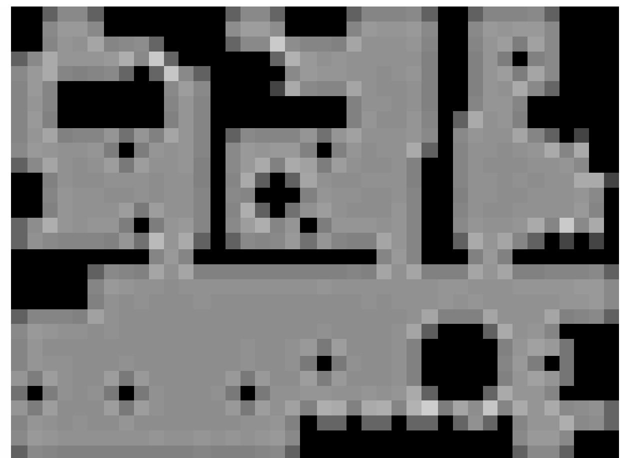
number of ant robots

Real-Time Search Methods

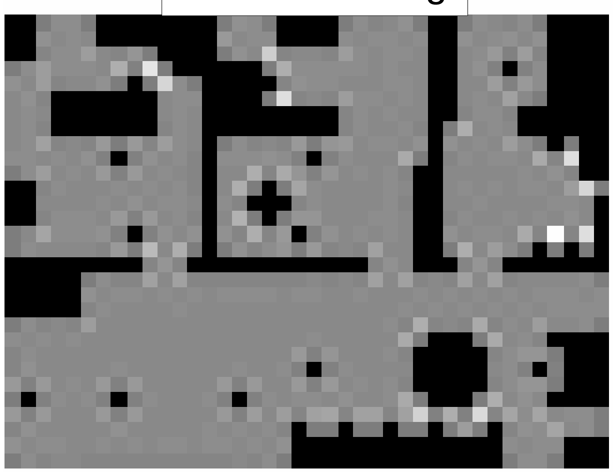
- How even is the coverage on average?



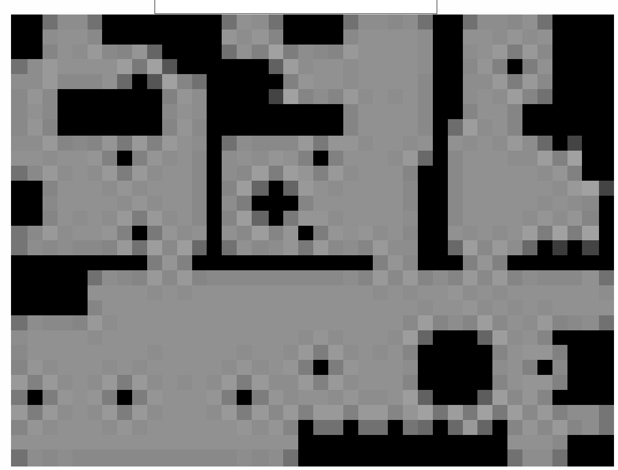
Node Counting



Korf’s LRTA*



Wagner’s Rule



Thrun’s Rule

Structure

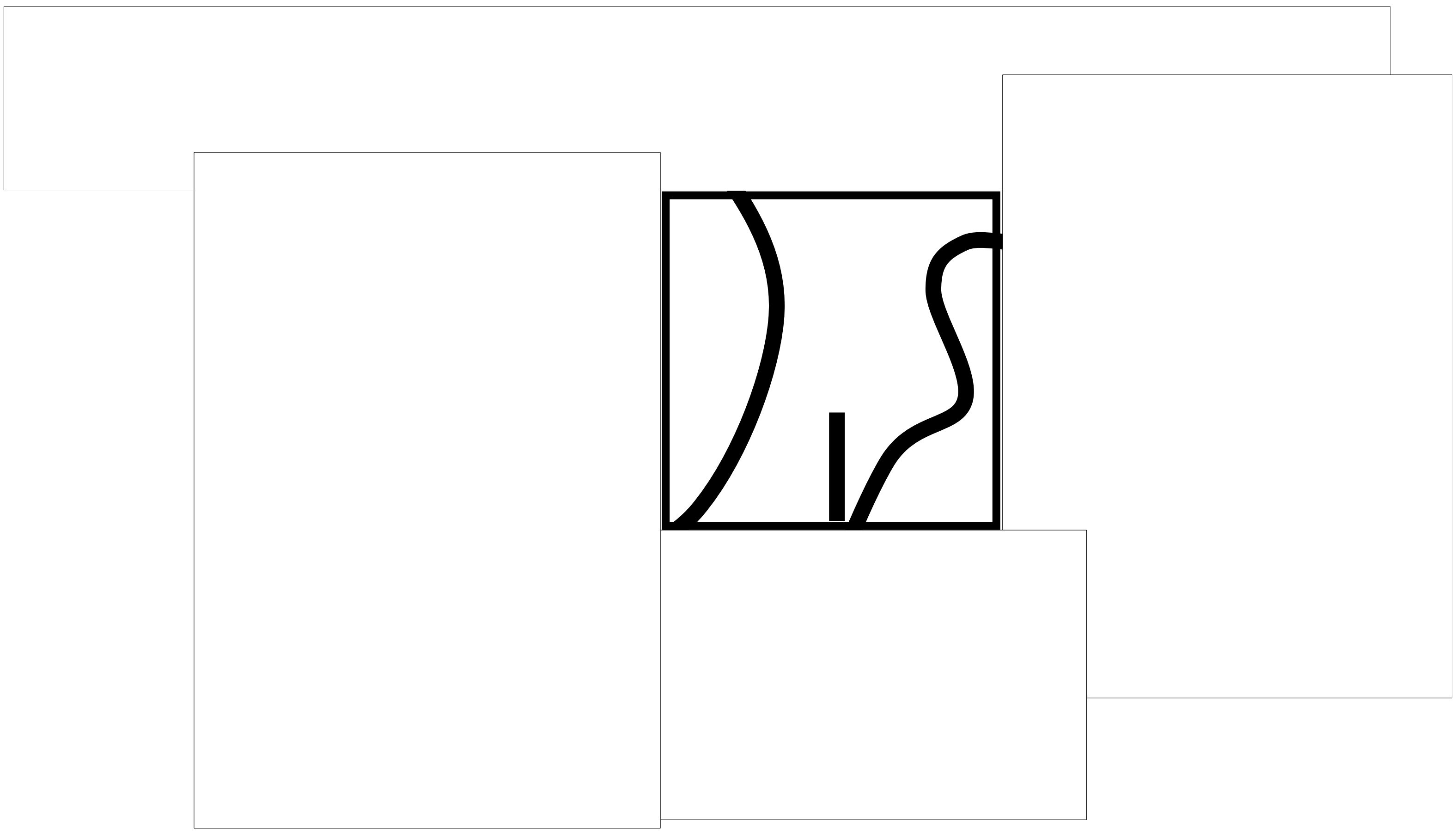
- Motivation
- Real-time search
- Results on real-time search
  - Analytical evaluation
  - Experimental evaluation
- Application to ant robots and results
- Serious application: smart markers

Real Robots and Simulations

- Ant robots that use Node Counting are easy to implement.

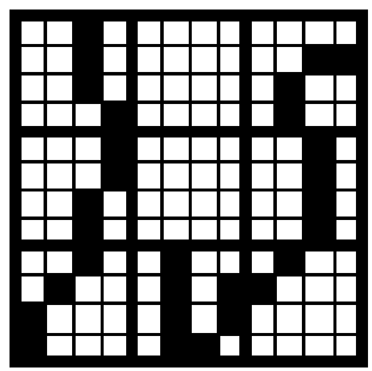
Real Robots and Simulations

- Ant robots that use Node Counting are easy to implement.



Real Robots and Simulations

- Ant robots that use Node Counting are easy to implement.



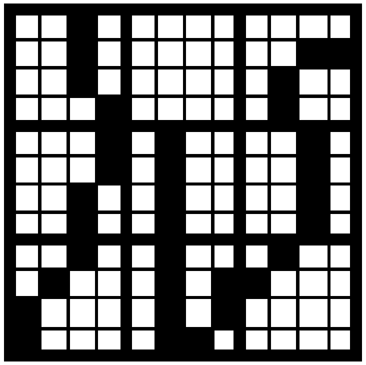
Real Robots and Simulations

- Ant robots that use Node Counting are easy to implement.

4	0	4
4	0	4
4	7	2

Real Robots and Simulations

- Ant robots that use Node Counting are easy to implement.



Real Robots and Simulations

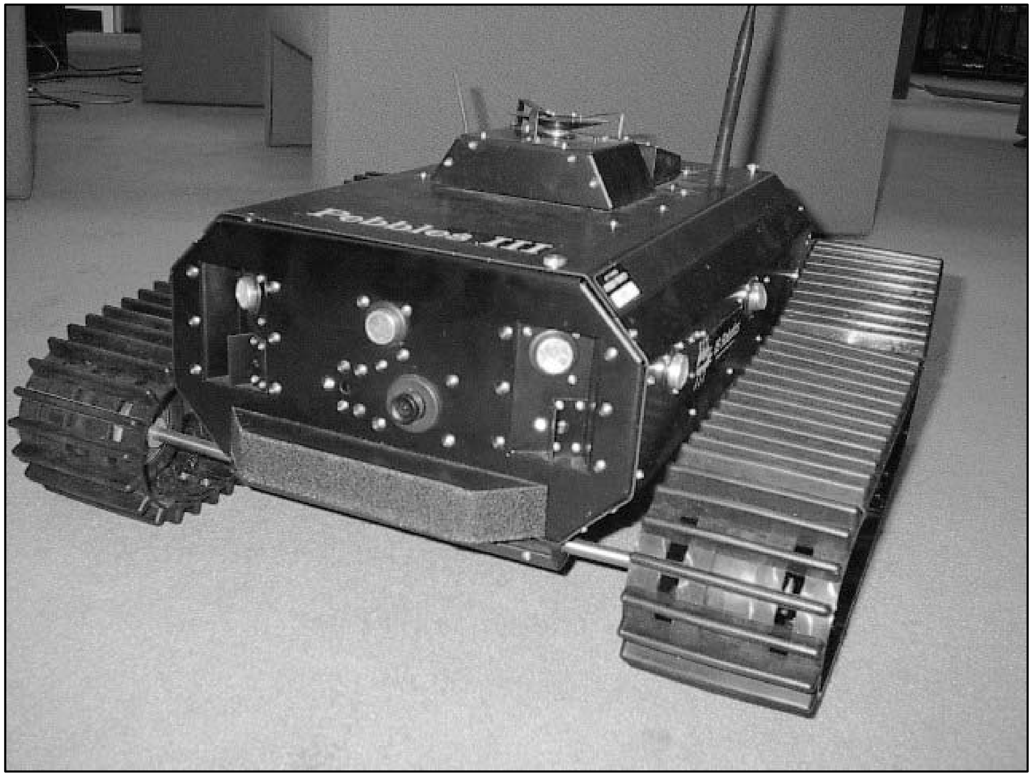
- Ant robots that use Node Counting are easy to implement.

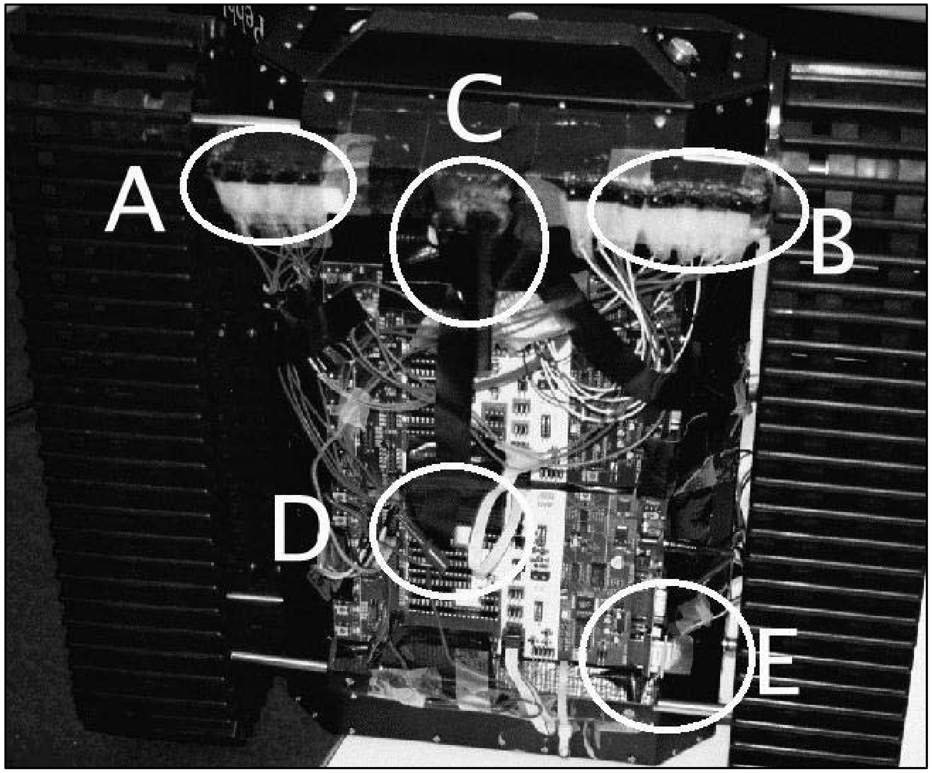
4	0	4
4	4	4
4	7	2

your system here

Real Robots and Simulations

- Ant robot hardware: Pebbles III

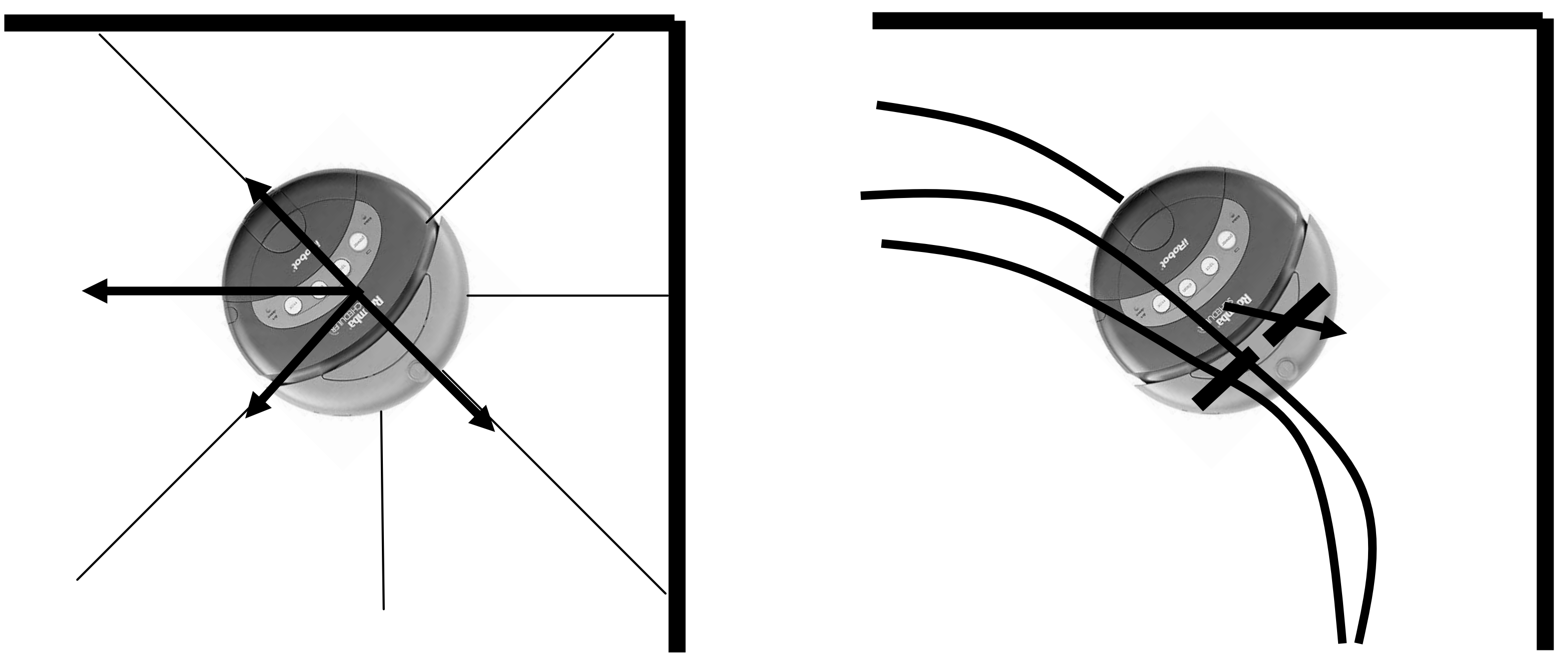




A: trail sensor, B: trail sensor, C: pen, D: micro-controller, E: RS232 interface

Real Robots and Simulations

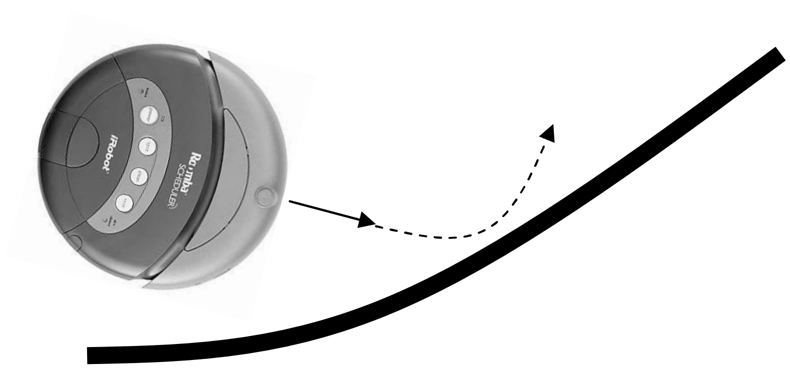
- Ant robot software: schema-based navigation

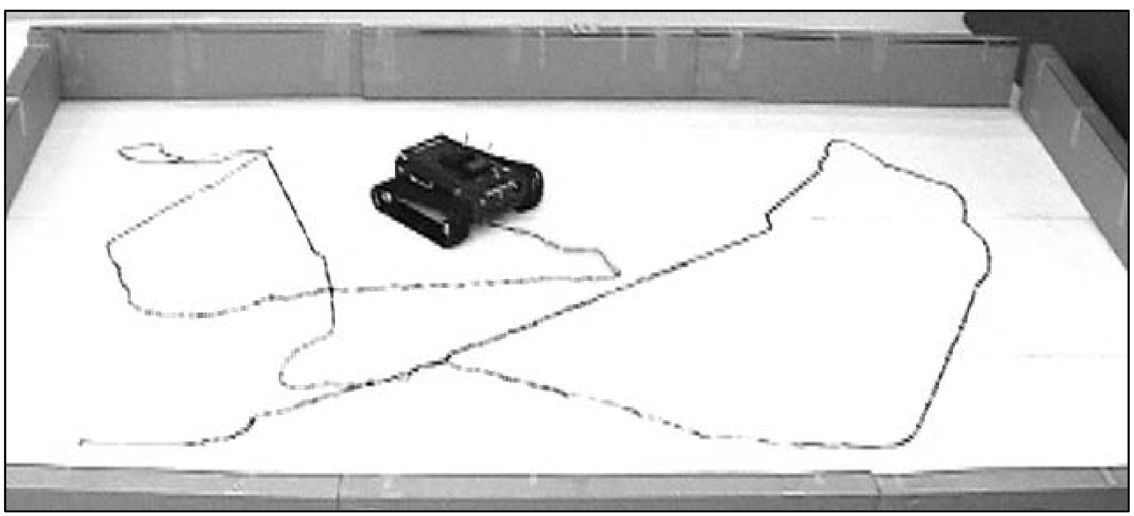


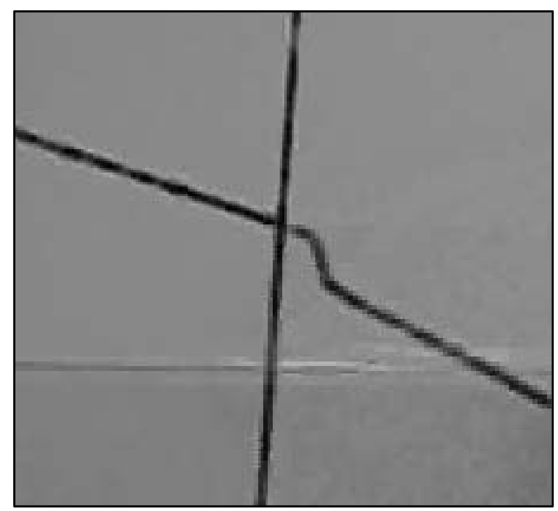
obstacle-avoidance behavior; trail-avoidance behavior

Real Robots and Simulations

- Ant robot software: importance of time delays

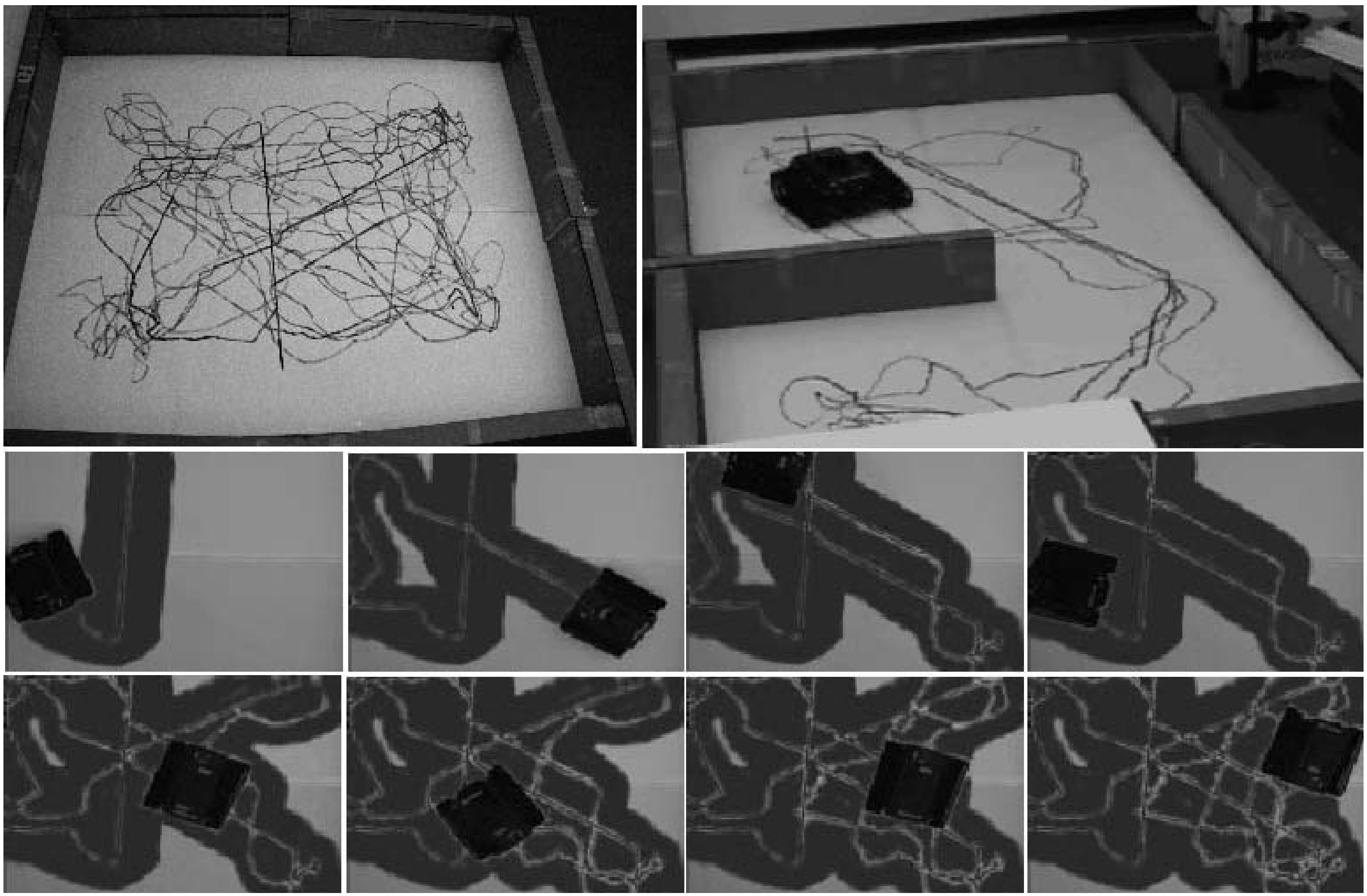






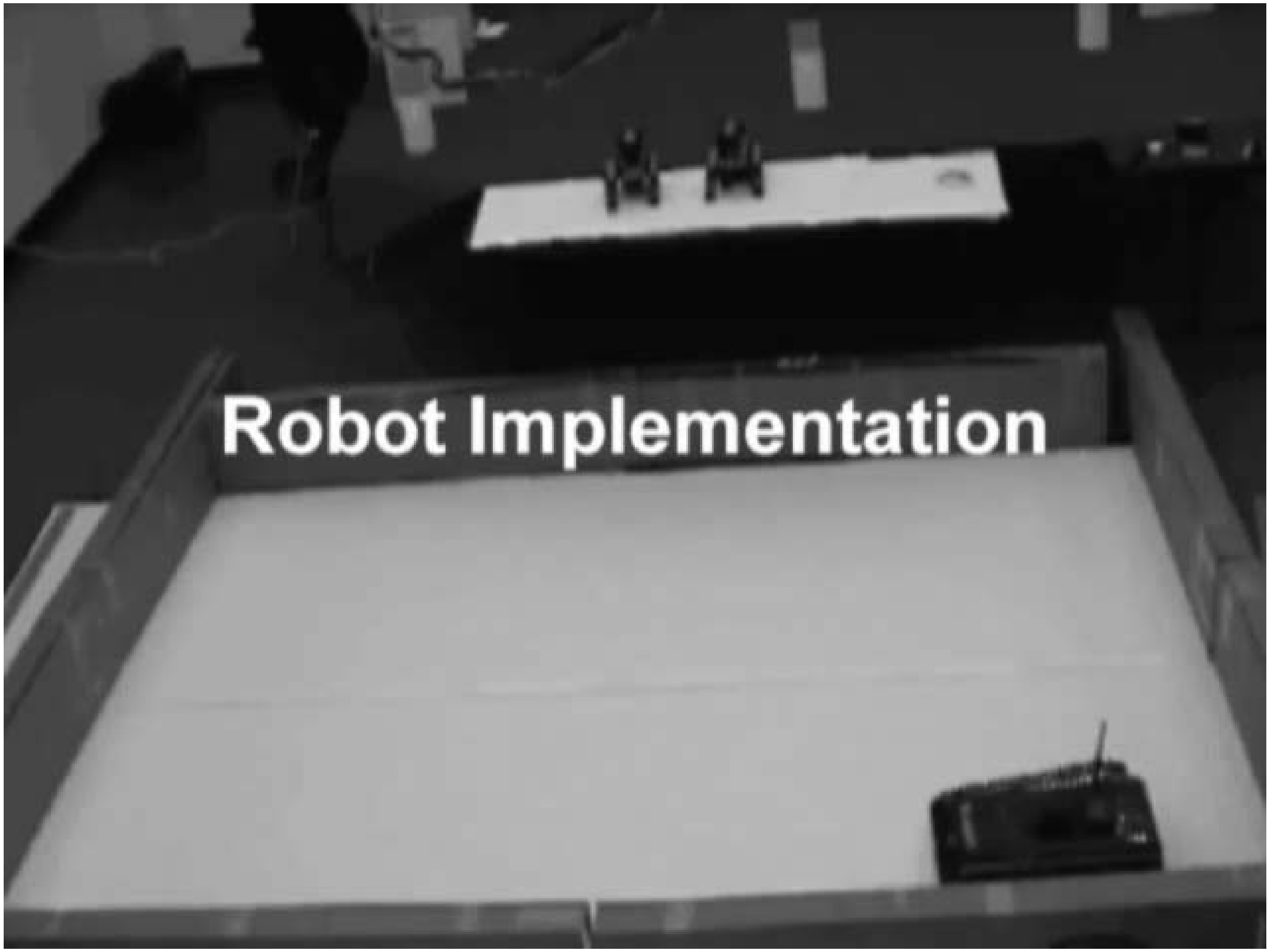
Real Robots and Simulations

- Empirical results



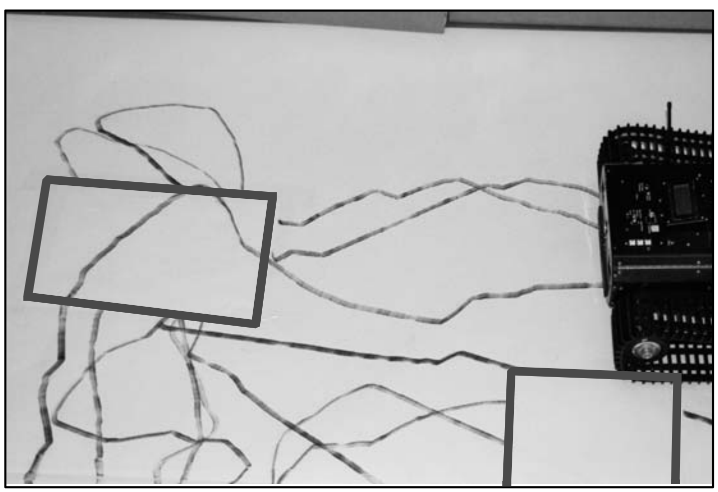
Real Robots and Simulations

- Empirical results

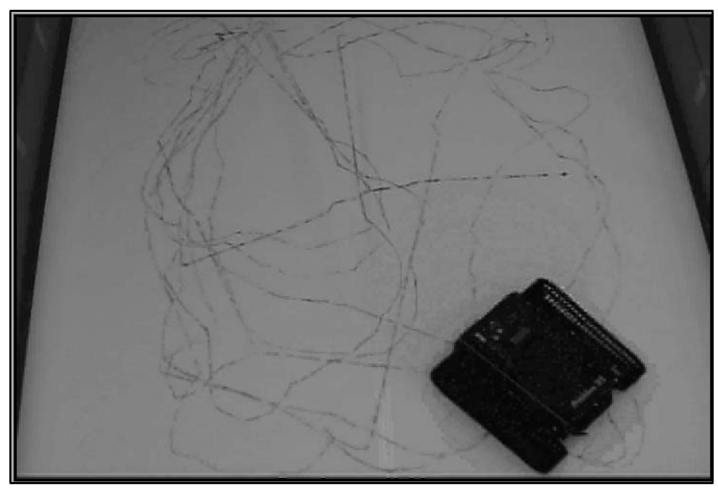


Real Robots and Simulations

- Our ant robots cover closed terrain even if
  - they don’t know the terrain in advance;
  - some ant robots fail;
  - some ant robots are moved without realizing this;
  - some trails are destroyed.



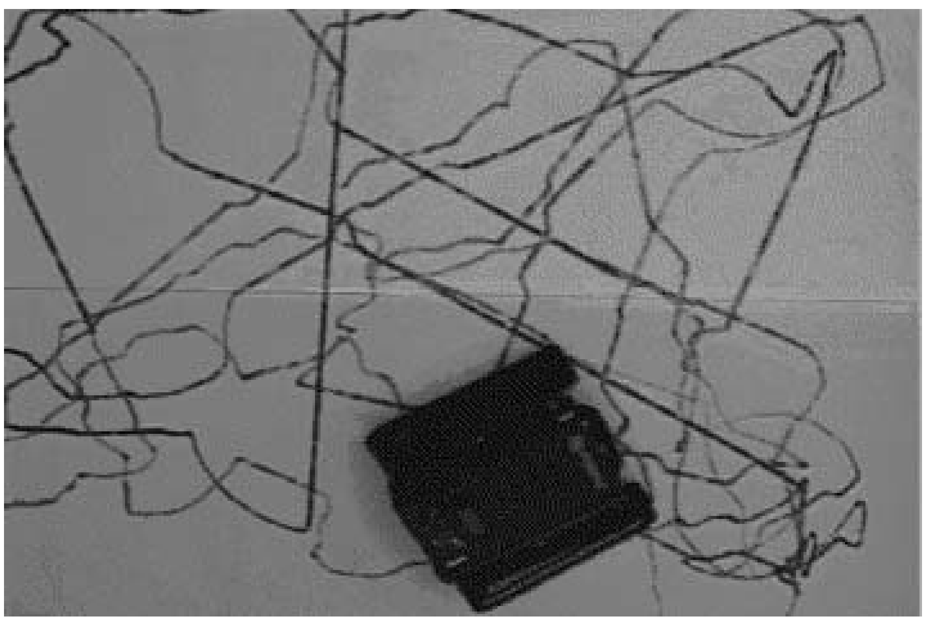
destroyed area of trails



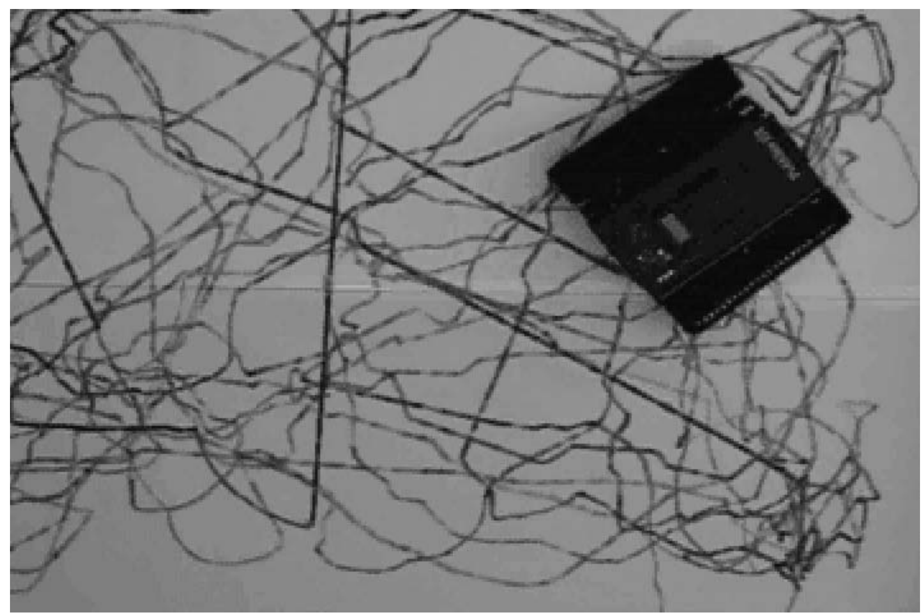
low-intensity trails

Real Robots and Simulations

- The terrain gets saturated with trails over time.

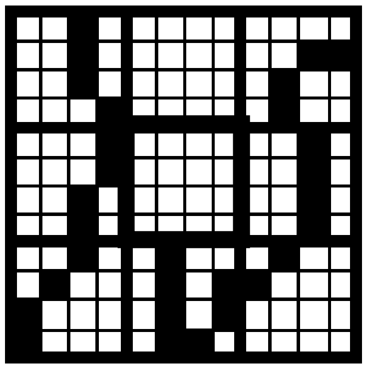


end of first coverage

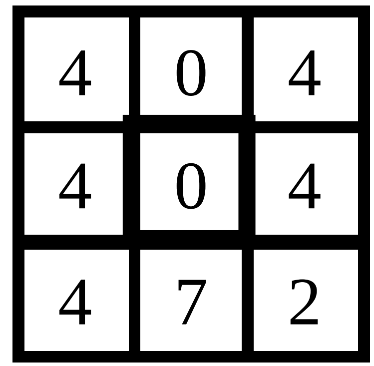


end of third coverage

Real Robots and Simulations



drop a drop of ink into a randomly chosen small cell in this large cell

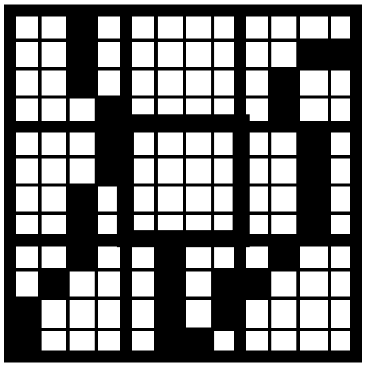


increase the u-value of this large cell by one with probability (16-0)/16

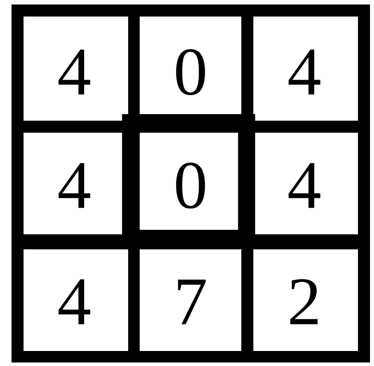
Initially, the u-values u(s) are zero for all cells s.

1. s := start cell
2. s' := a cell adjacent to cell s with a minimal u-value
3. u(s) := 1 + u(s)
4. move the ant robot to cell s'
5. go to 2

Real Robots and Simulations



drop a drop of ink into a randomly chosen small cell in this large cell

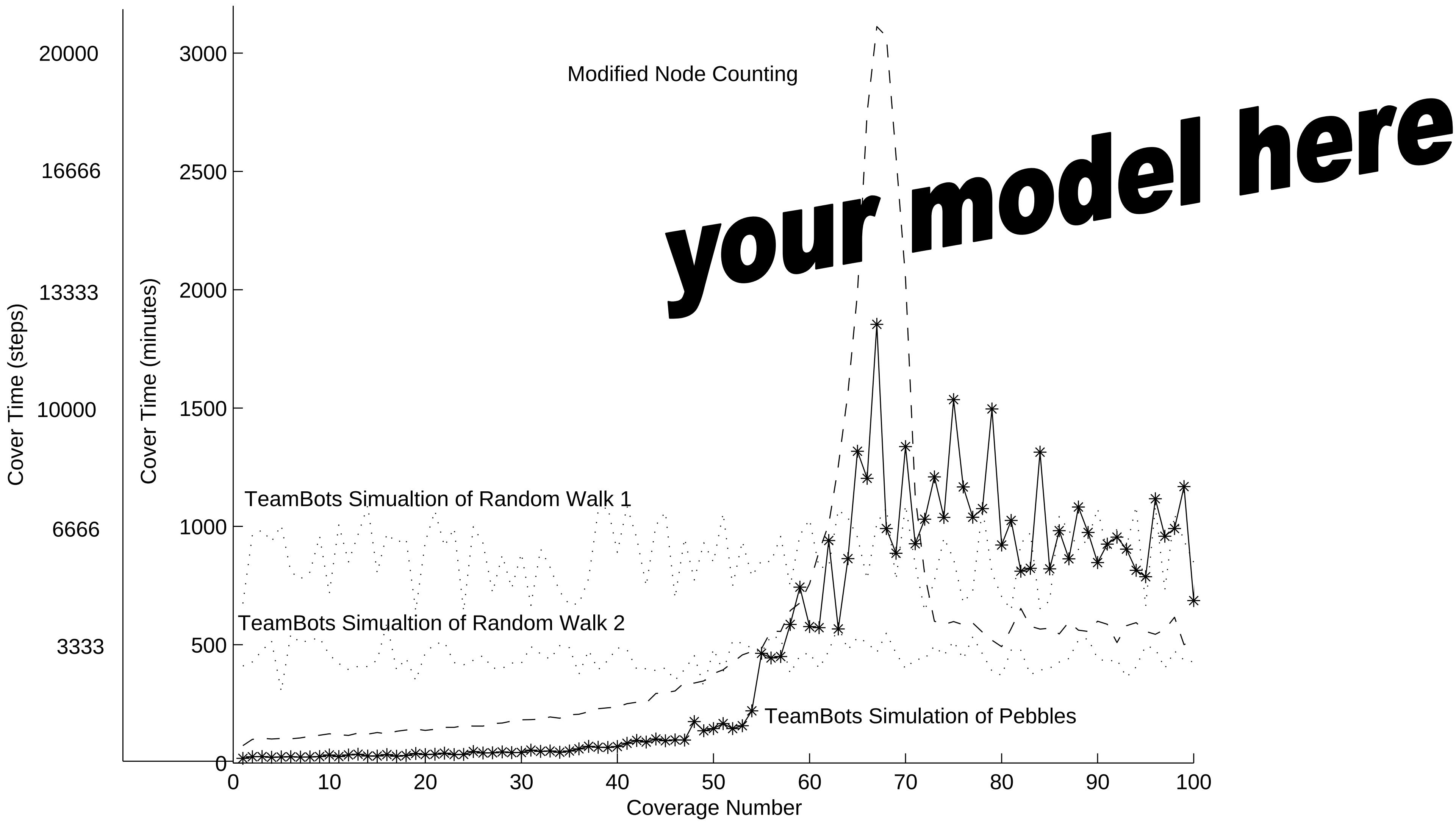


increase the u-value of this large cell by one with probability (16-0)/16

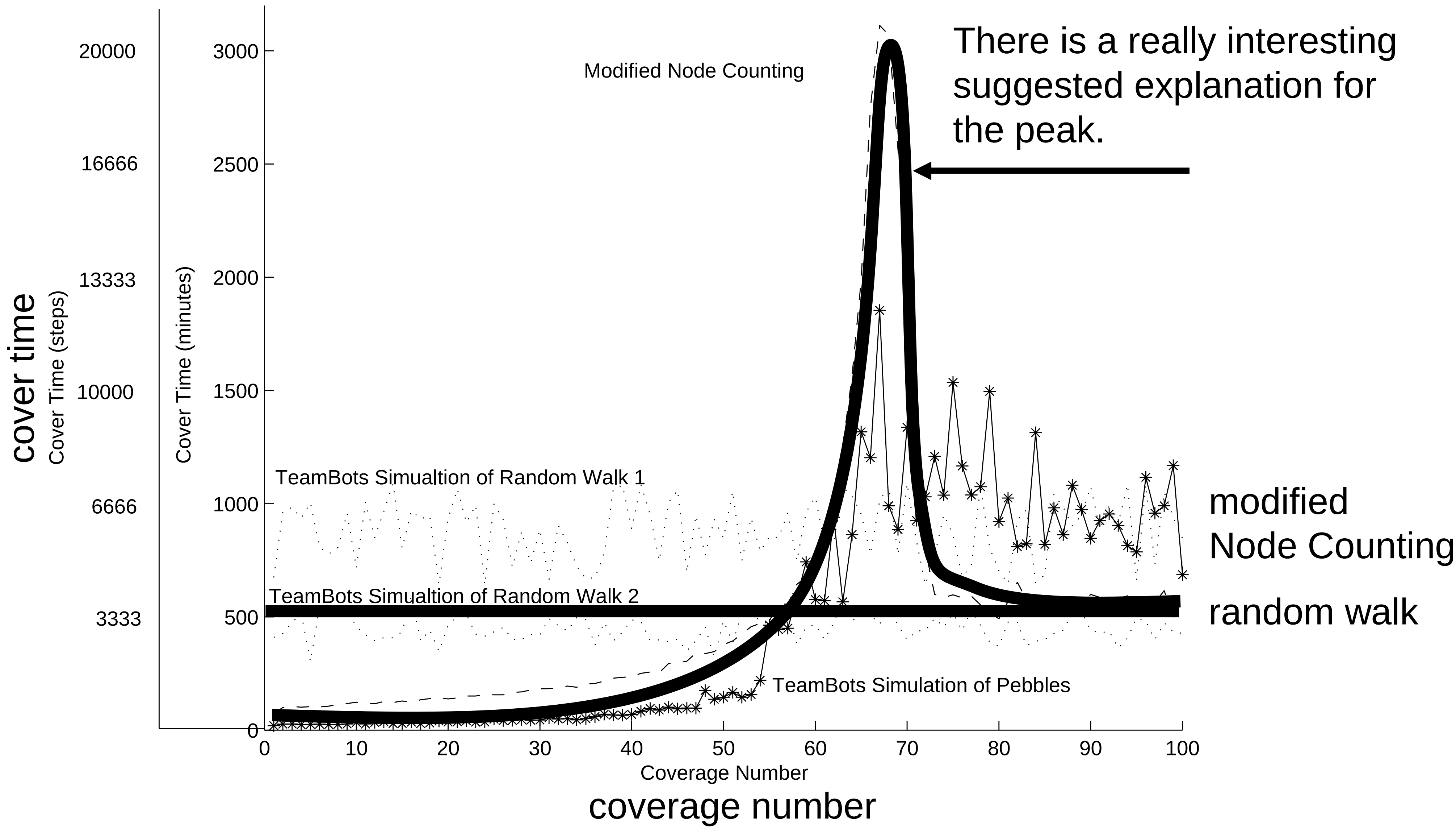
Initially, the u-values u(s) are zero for all cells s.

1. s := start cell
2. s’ := a cell adjacent to cell s with a minimal u-value
3. with probability (170-u(s))/170 do: u(s) := 1 + u(s)
4. move the ant robot to cell s’
5. go to 2

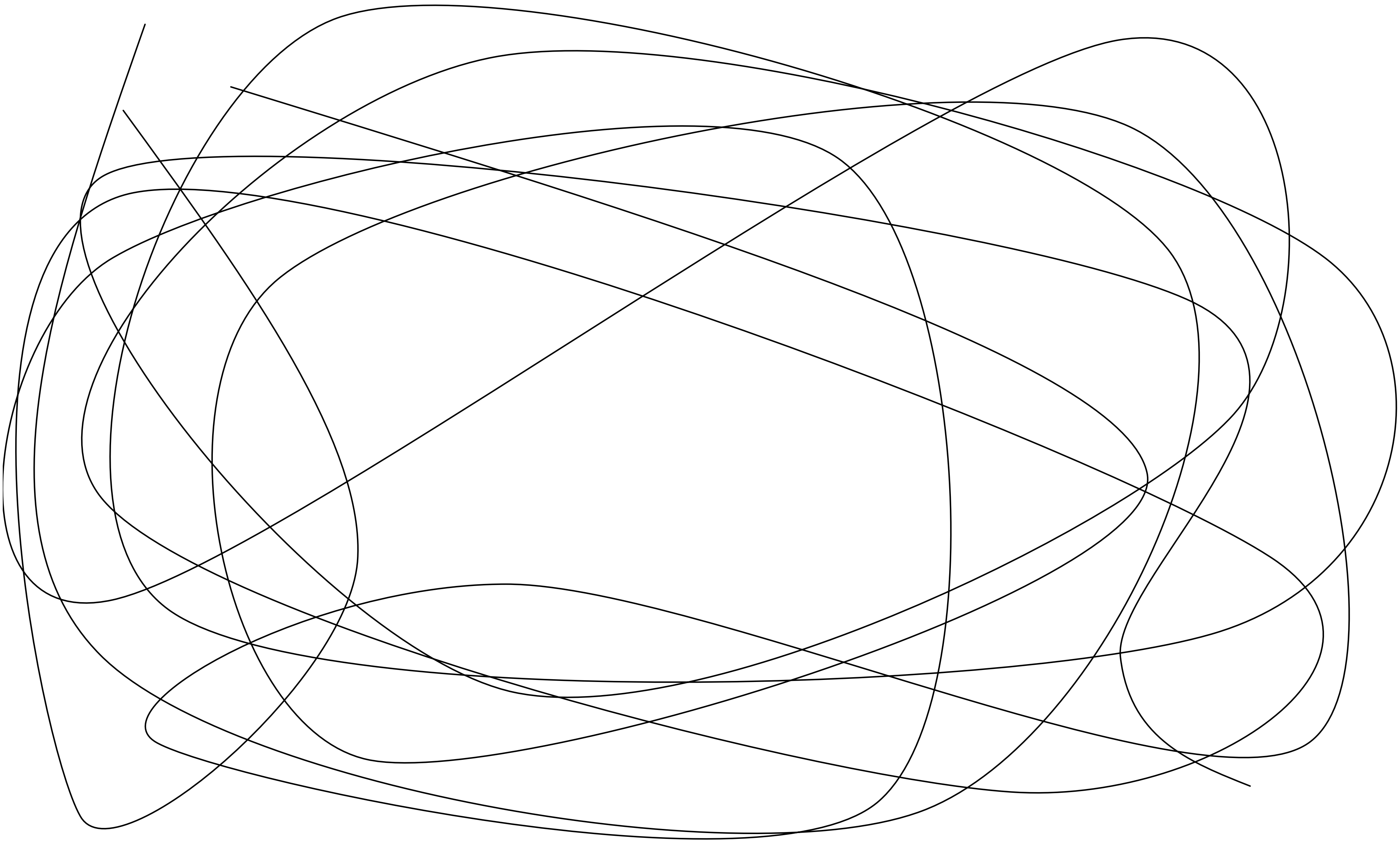
Real Robots and Simulations



Real Robots and Simulations



Real Robots and Simulations



Real Robots and Simulations

- The terrain gets saturated with trails over time. We can avoid this by
  - letting the trails evaporate
    - but the evaporation rate depends on the terrain size
  - removing the trails (via cleaning)
    - but we need to avoid odd behaviors
      - e.g. one robot repeatedly turning around forever
      - e.g. one robot following another robot forever

you letting trails evaporate here

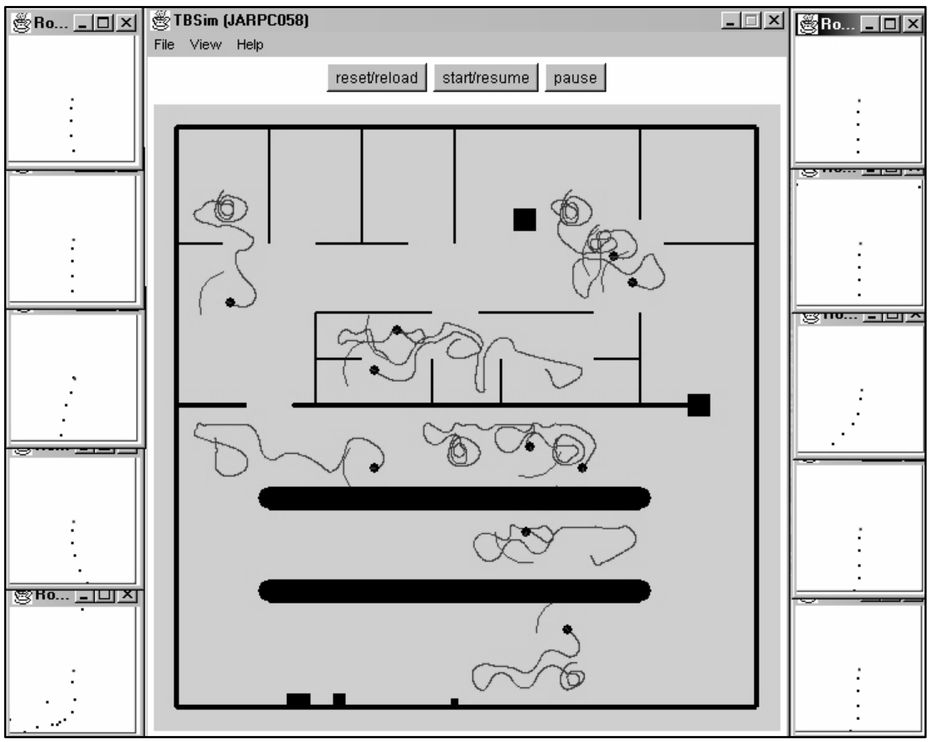
Real Robots and Simulation

- We chose to remove the trails.

your removal design here

Real Robots and Simulators

- 85 hours without any ant robot getting stuck in a realistic robot simulator

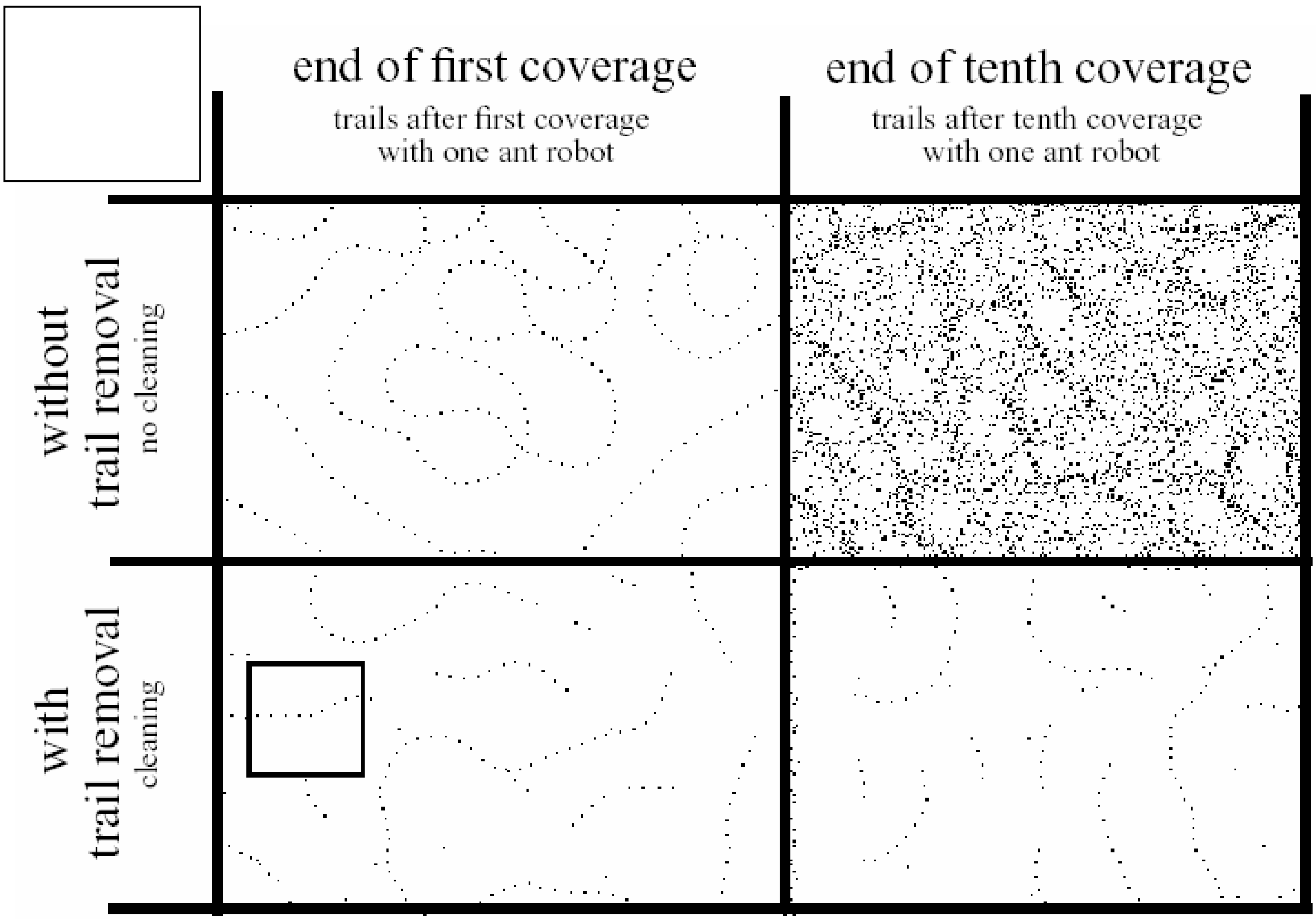


10 ant robots in a 25 by 25 meter terrain

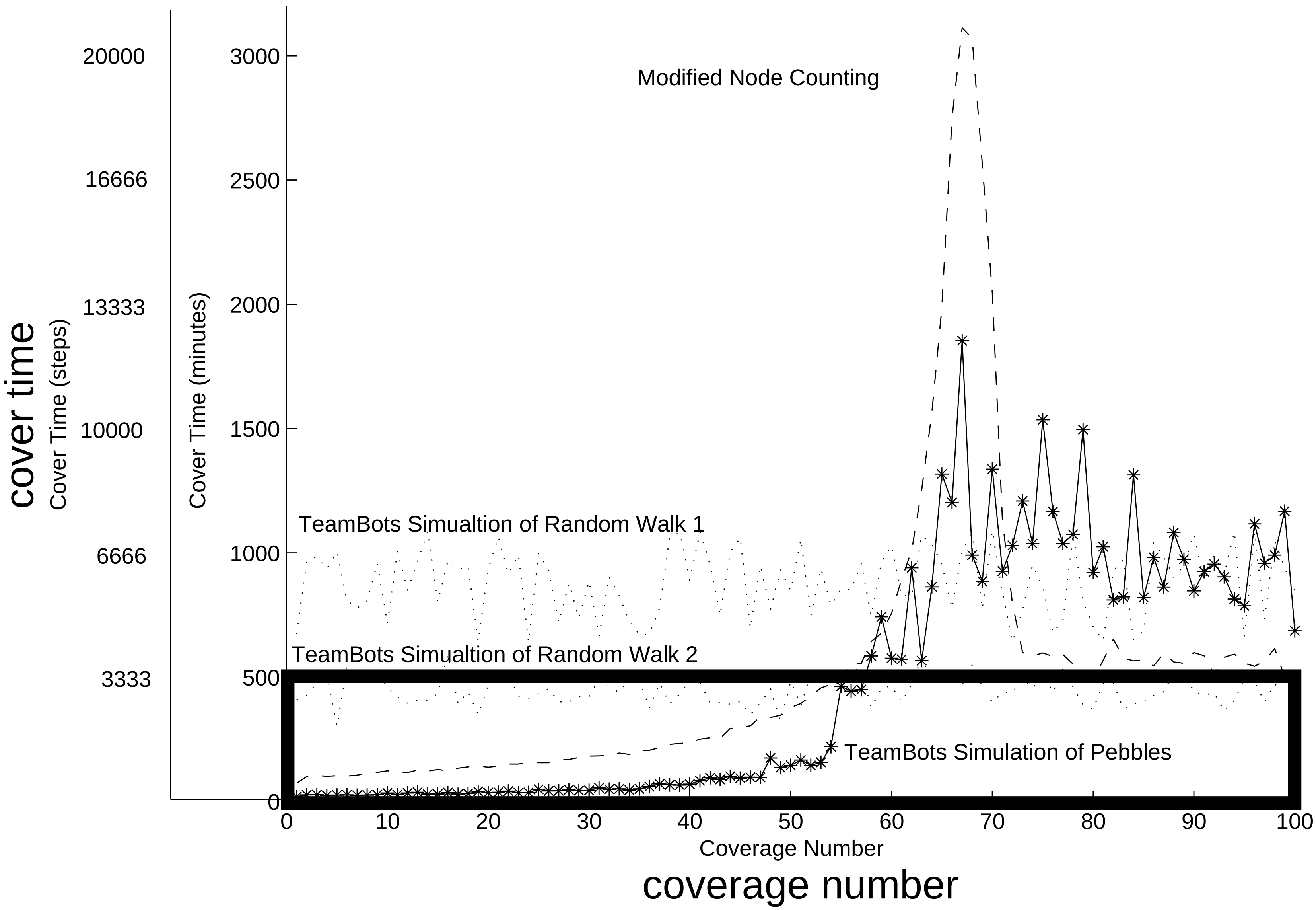
Real Robots and Simulators

- 85 hours without any ant robot getting stuck in a realistic robot simulator

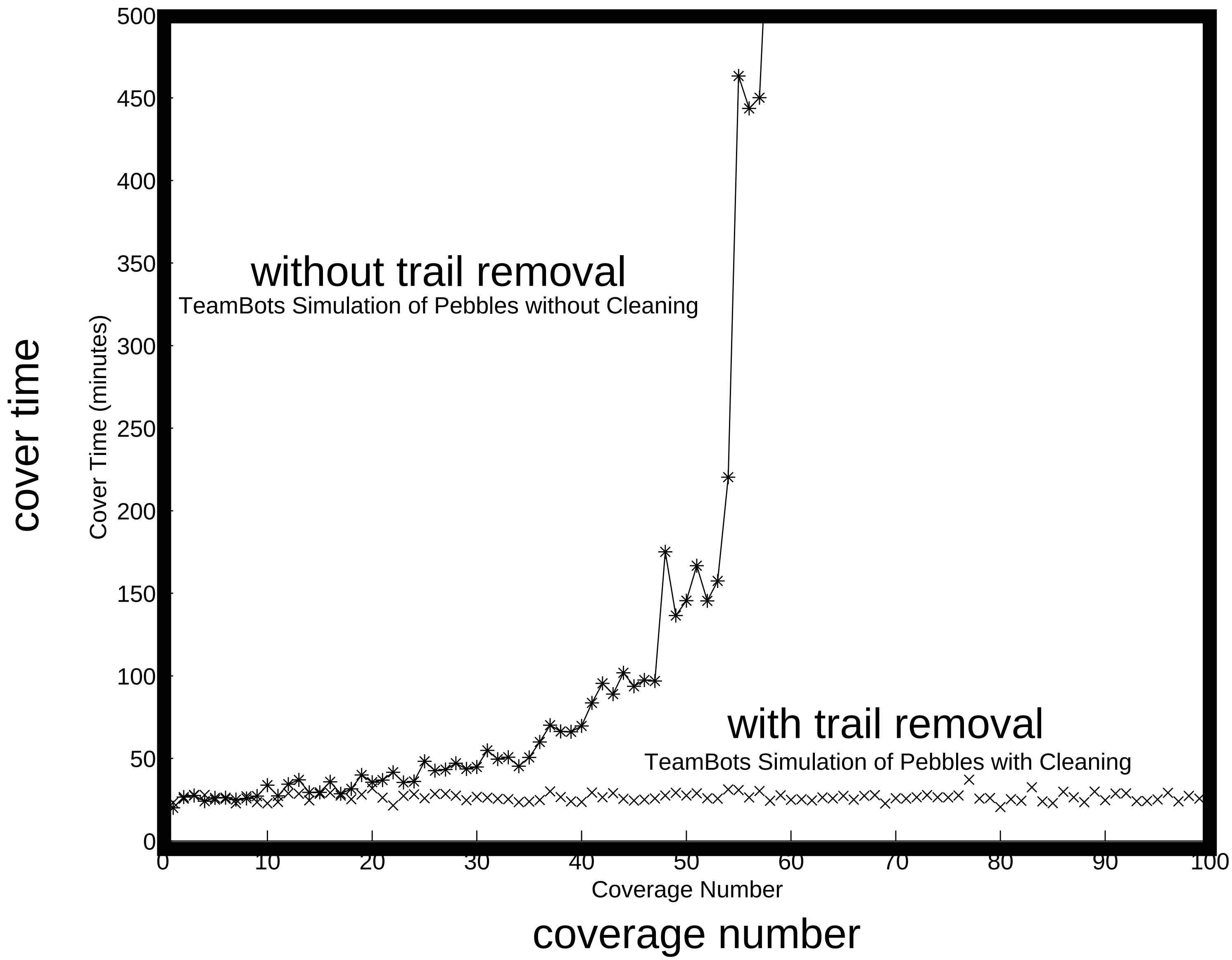
Real Robots and Simulators



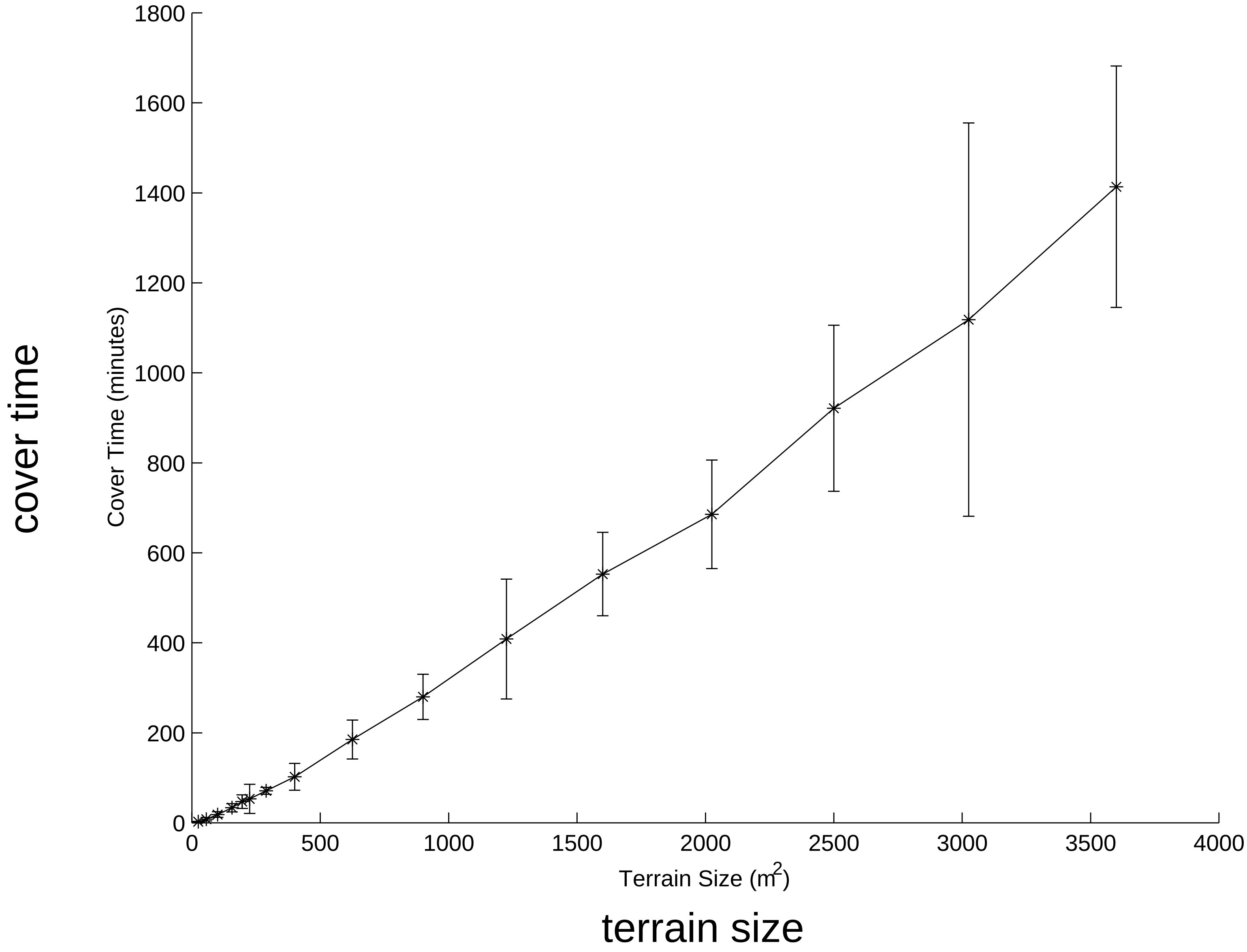
Real Robots and Simulations



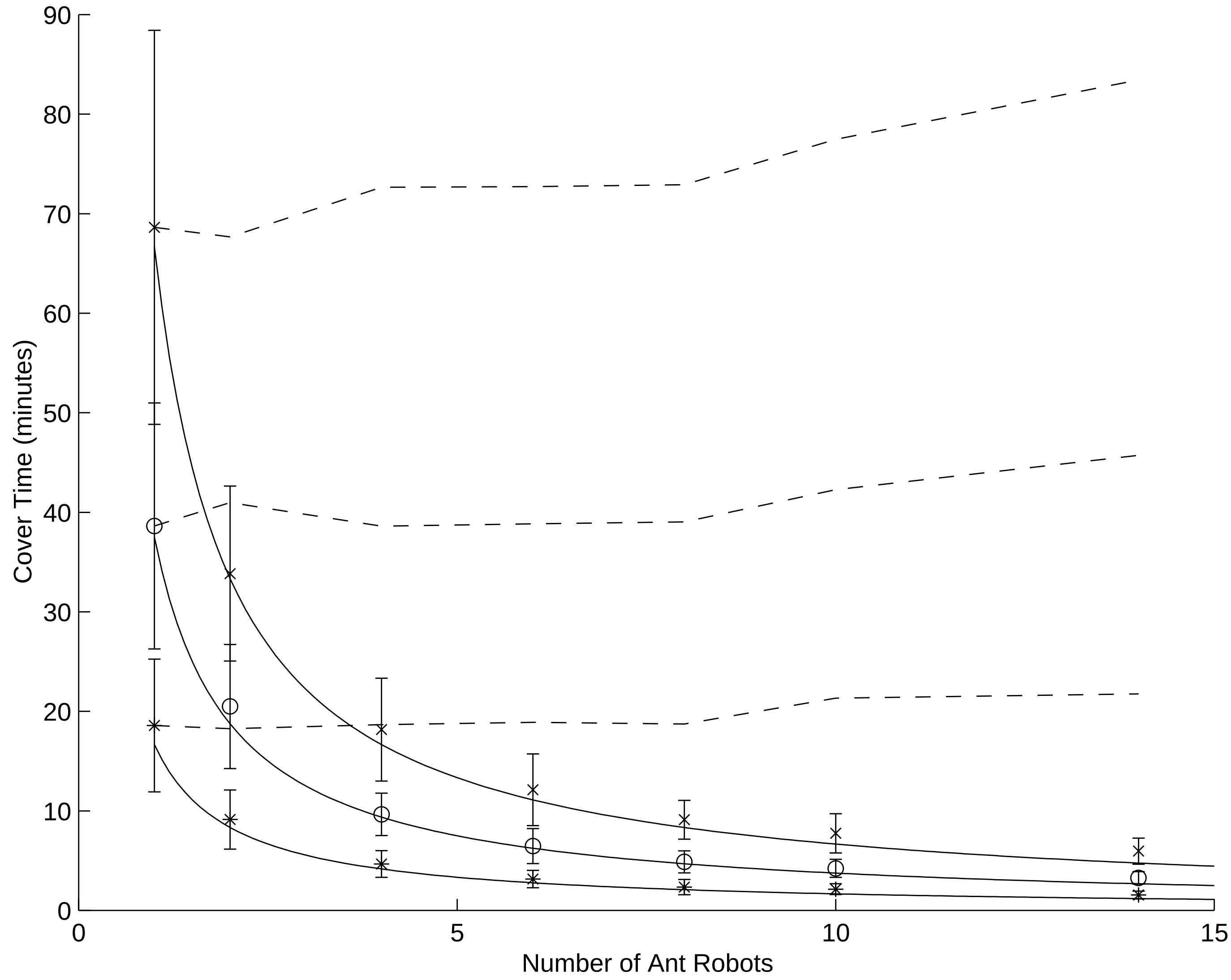
Real Robots and Simulations



Real Robots and Simulations



Real Robots and Simulations



cover time

number of ant robots

Structure

- Motivation
- Real-time search
- Results on real-time search
- Application to ant robots and results
  - Analytical evaluation
  - Experimental evaluation
- Serious application: smart markers

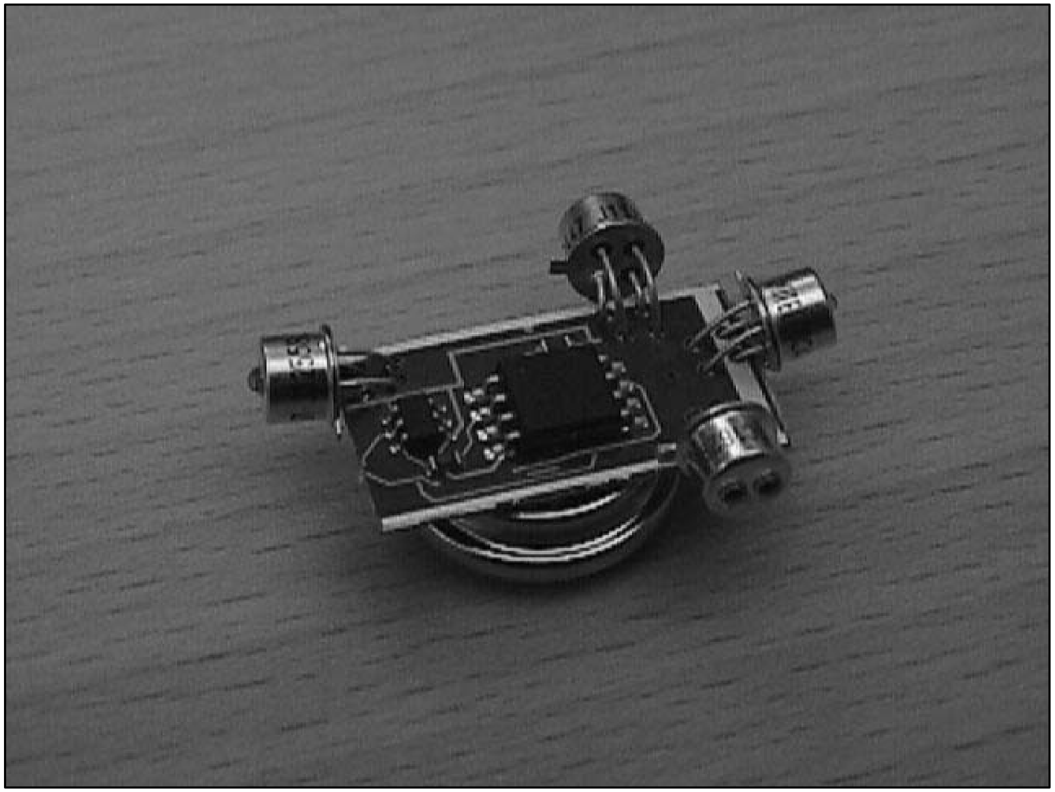
Motivating Toy Task

- Guarding a museum at night?
  - Why contaminate the terrain with trails?
    - People might slip.
    - The trails might be toxic.
    - The trail substance might be expensive.
  - Why not simply use a system of cameras?

your application here

- See whether you can come up with a good application in the following. We use search and rescue after an earthquake.

Smart Markers

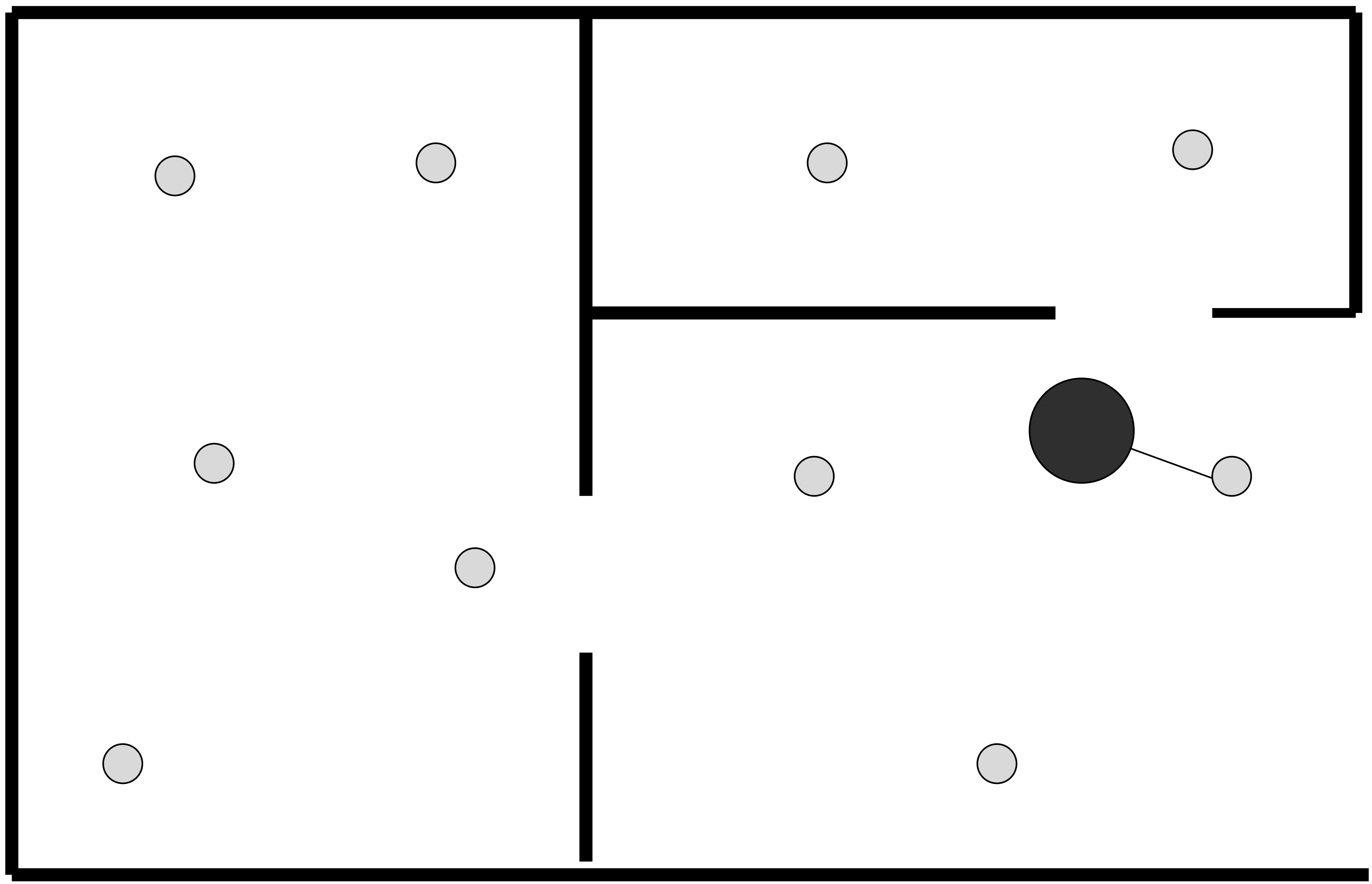


small infrared transceivers (eventually about 1 dollar each)

Robots are now localized with respect to the closest transceiver.

We don’t avoid the localization problem, we solve it!

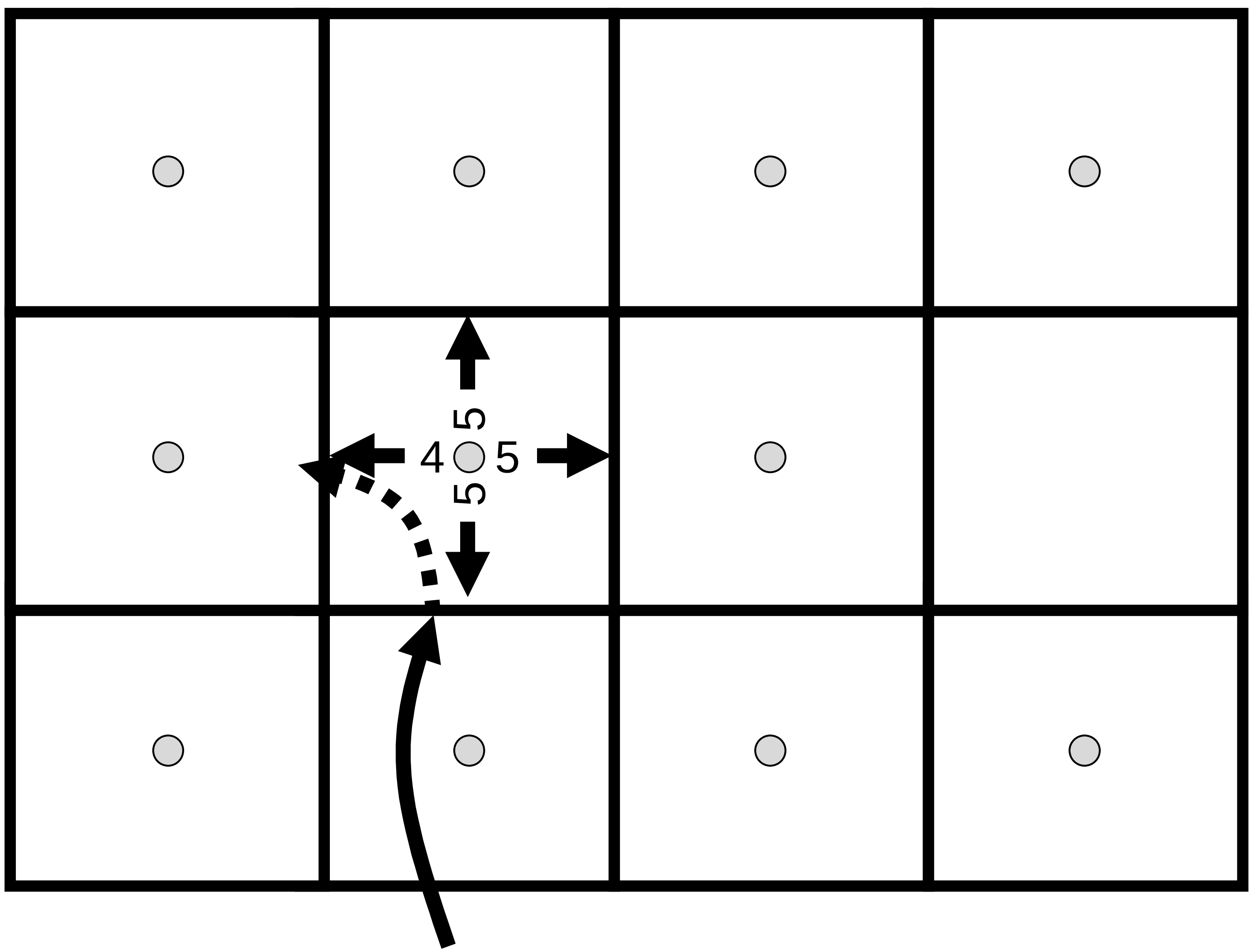
Isolated Smart Markers



Isolated Smart Markers

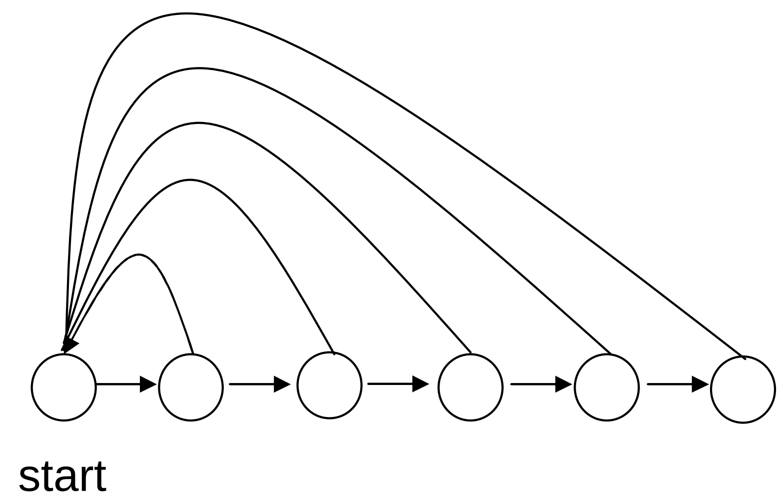
- A robot always moves in the direction in which the closest smart marker has been left the smallest number of times (= Edge Counting = Edge Ant Walk).

Edge Counting

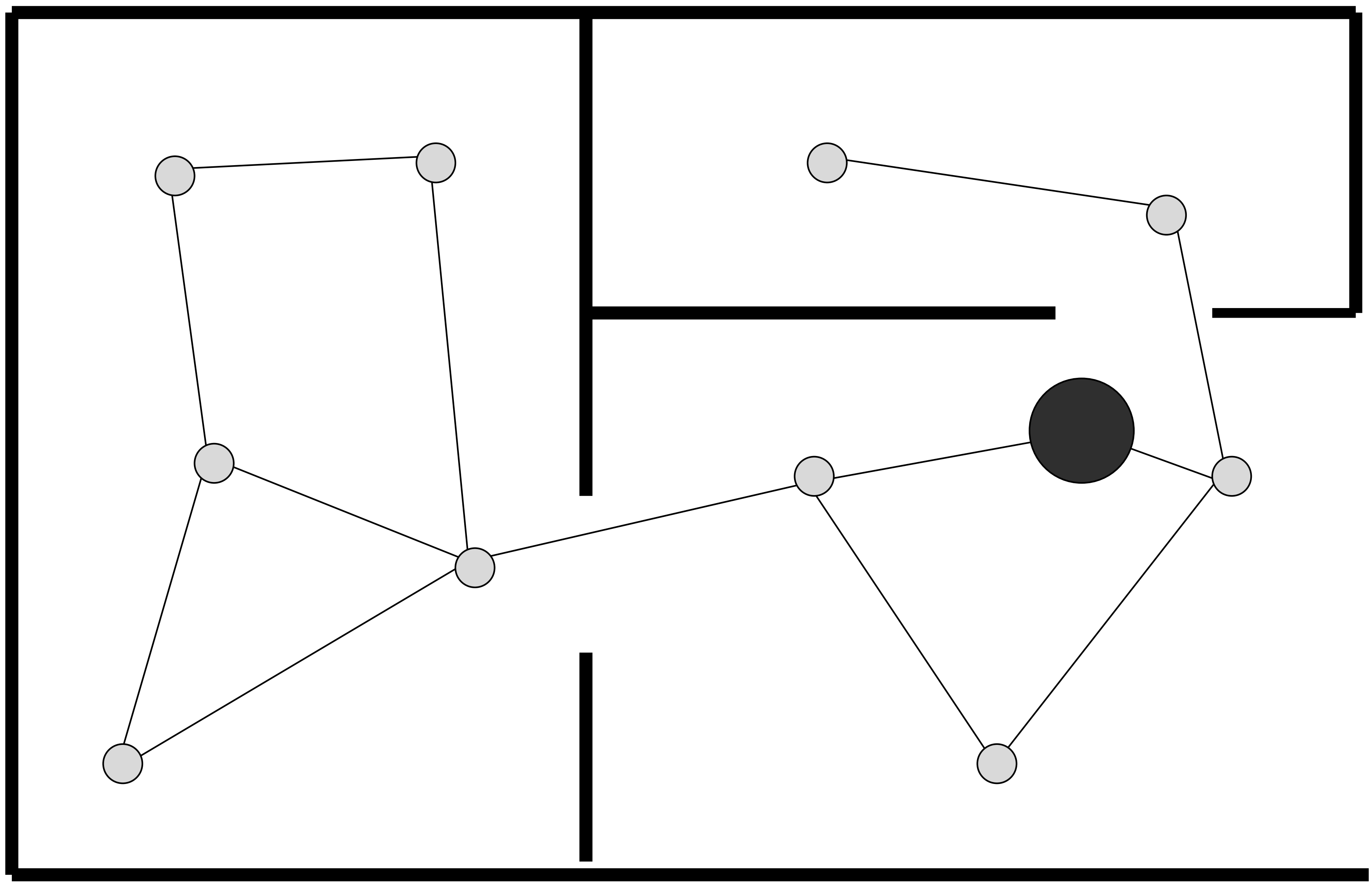


Edge Counting

- The worst-case cover time of Edge Counting is exponential on strongly connected graphs. (The example is similar to the one for Node Counting.) It is at most #edges × diameter movements on undirected or Eulerian graphs, including grids. [Koenig]



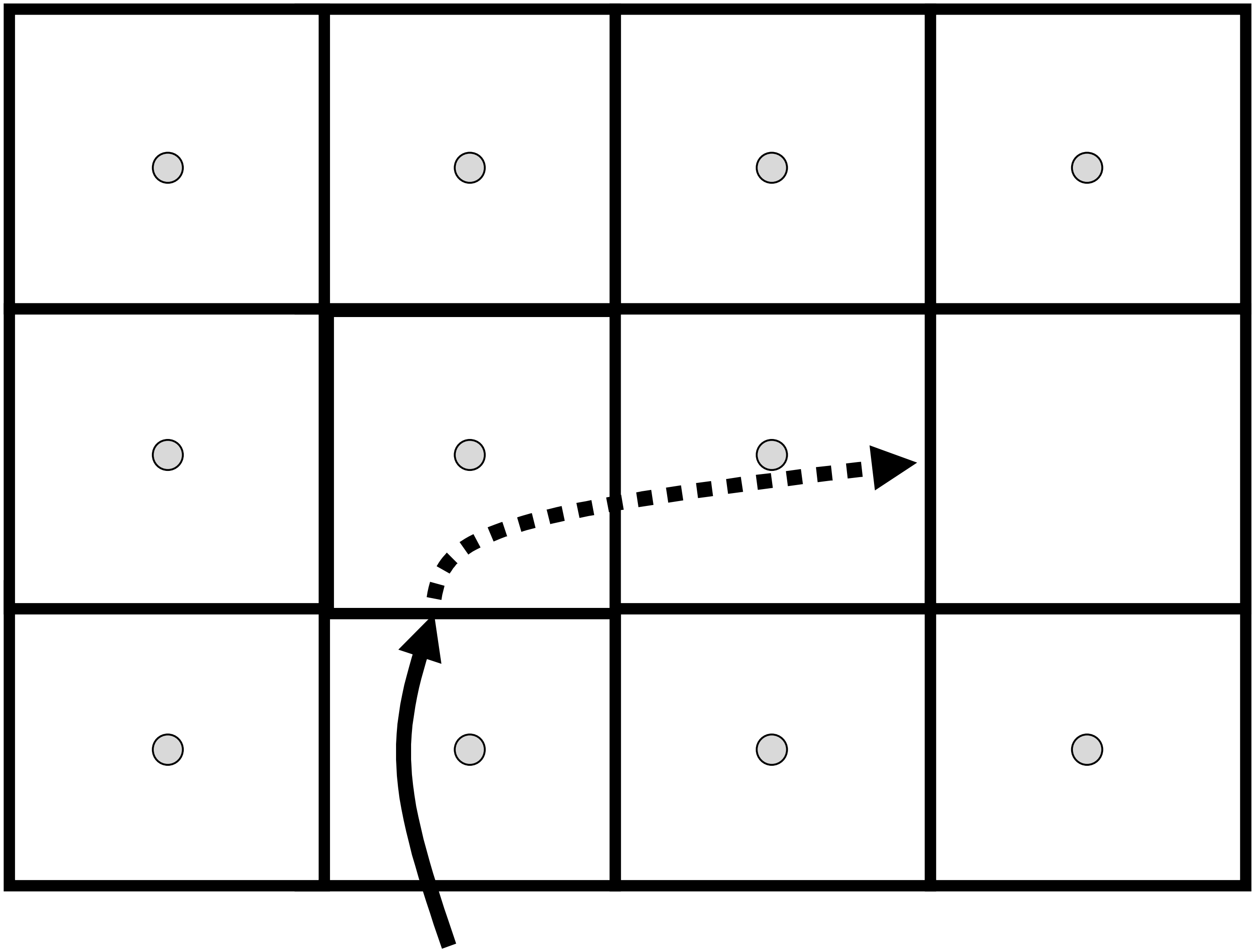
Networks of Smart Markers



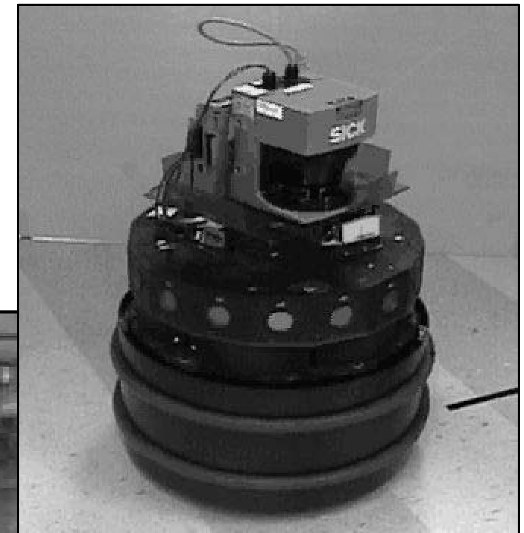
Networks of Smart Markers

- A robot always moves in the direction in which there is no smart marker yet (= Greedy Mapping).

Greedy Mapping



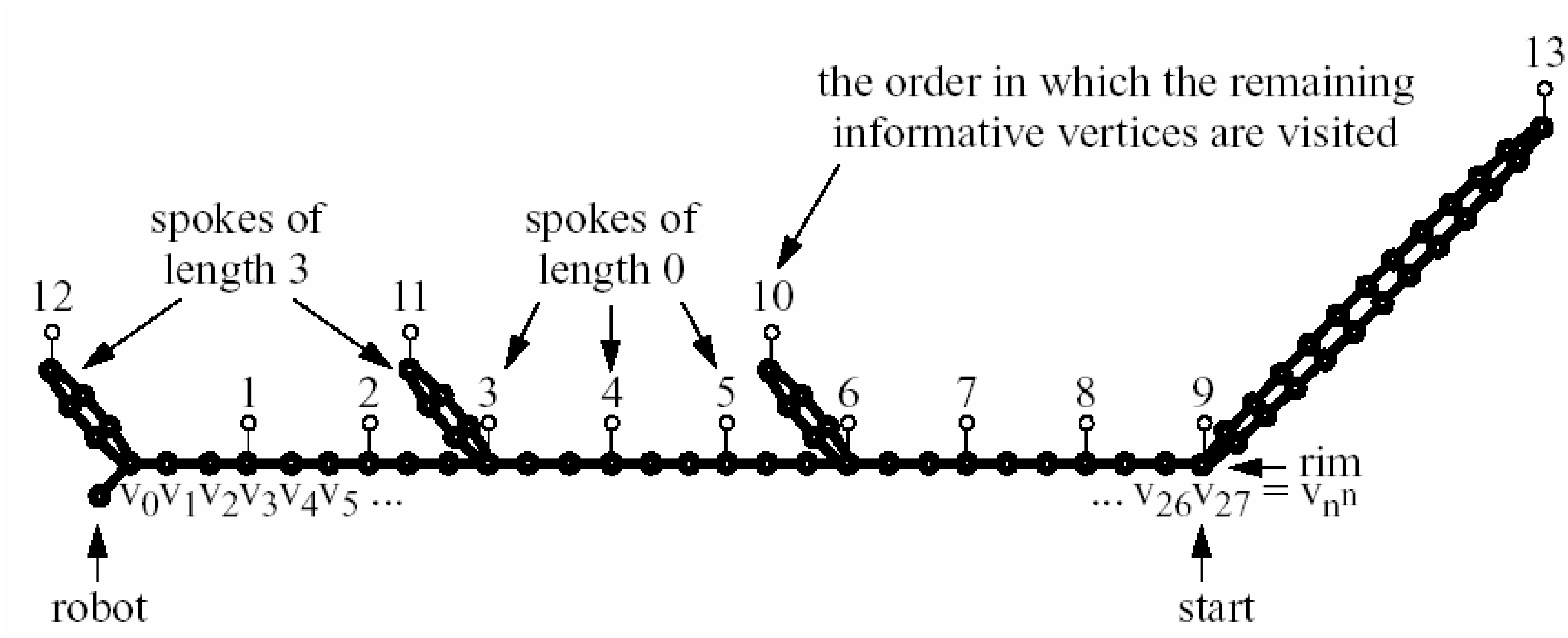
Greedy Mapping



your system and analysis here

Greedy Mapping

- The worst-case cover time of Greedy Mapping is Ω((log #vertices / log log #vertices) #vertices) and O(#vertices log #vertices) movements on undirected connected vertex-blocked graphs, even on grids. [Tovey et. al.]



Further Information

- This was a short version of the tutorial on ant robotics. Additional information about my own research on ant robotics can be found at:

  idm-lab.org/project-b.html.

- Thank you!