ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Distributed Intelligent Systems – W13 and W14:

Distributed Sensing using Robotic Sensor Networks

Outline

- **Robotic sensor networks: challenges and applications**
- **Robotic sensor networks for distributed search: a case study on odor source localization**
  - **Challenges and approach**
  - **Single robot algorithms**
  - **Multi-robot algorithms**
- Robotic sensor networks for distributed coverage: a case study on turbine inspection
  - Challenges and approach
  - Single robot algorithms
  - Multi-robot algorithms

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

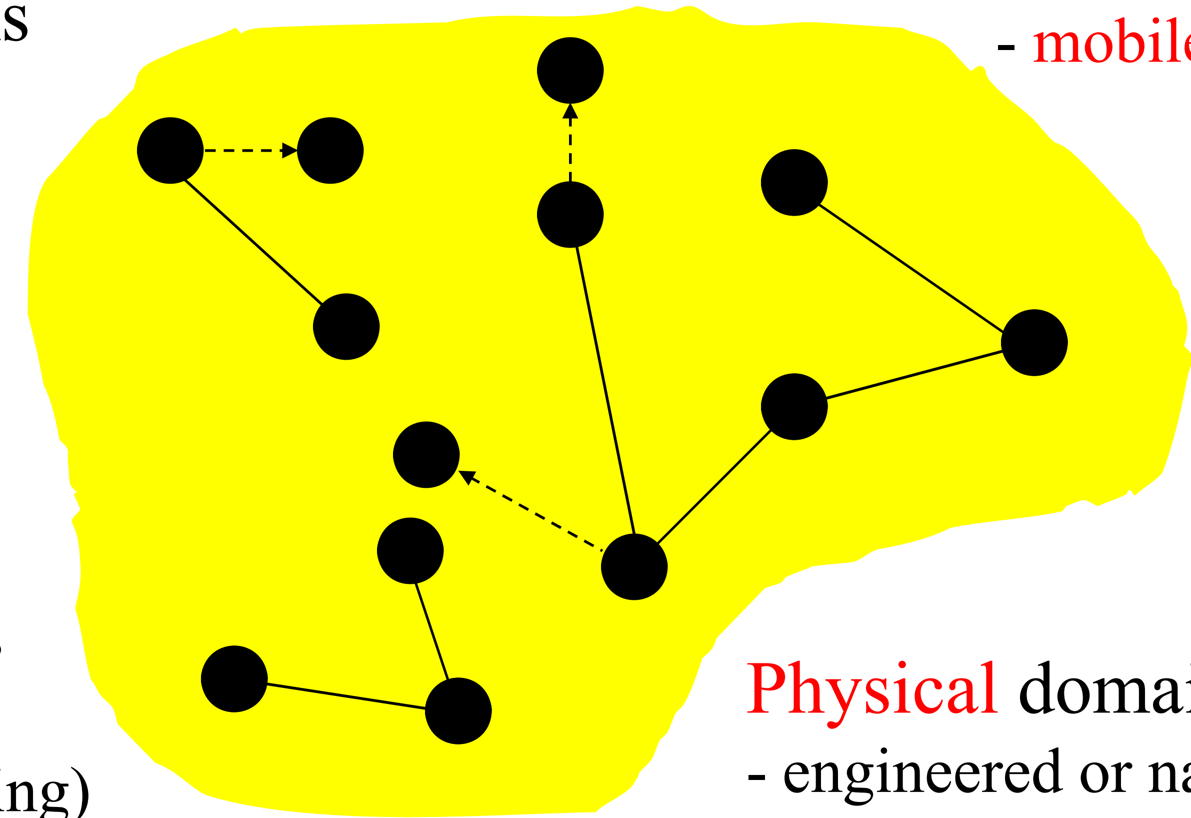
Robotic Sensor Networks

Distributed Sensing: Problem Statement

Devices: - size, cost
- number
- networked
- mobile

Possible missions
- searching
- coverage
- inspecting
- patrolling
- mapping
- …

+ follow-up actions (e.g., cleaning, destroying, repairing)



Physical domain
- engineered or natural
- bounded or unbounded
- 2D or 3D

Typical Performance Metrics

- Coverage: spatial exhaustiveness > time of completion (i.e. minimize redundant coverage but better redundancy than uncovered areas)
- Search: time of completion > spatial exhaustiveness (speed is the most important issue)
- Depending on a priori knowledge (e.g., target features, # of targets) some overlap possible
- Both problem classes should be solved by minimizing system resources (e.g., energy, # of robots, node cost)

Application Examples for Robotic Sensor Networks

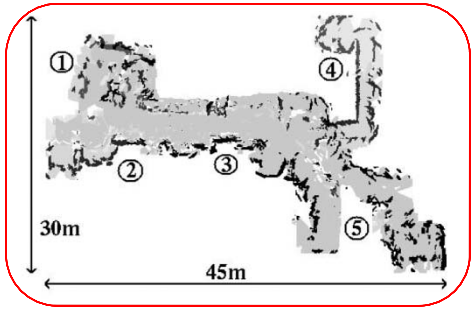
Typical Applications

- Cleaning, mowing
  (Jaeger & Nebel 2002)
- Communication backbone
  (McLurkin & Smith 2006)
- Guiding facility for other robots
  (Payton, Estkowski & Howard 2003)
- Distributed sensor
  (Schwager, McLurkin & Rus 2006)
- Mapping
  (Zlot, Stentz, Dias & Thayer, 2002), (Burgard, Moors, Stachniss & Schneider, 2005)
- Urban search and rescue
  (Kumar, Rus & Singh 2004)

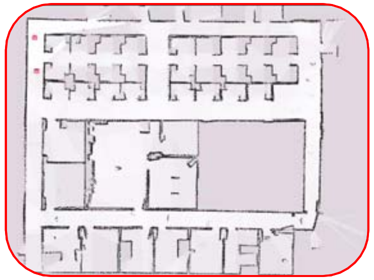
© James McLurkin, MIT/iRobot

© Husquarna

© Kumar, Rus & Singh



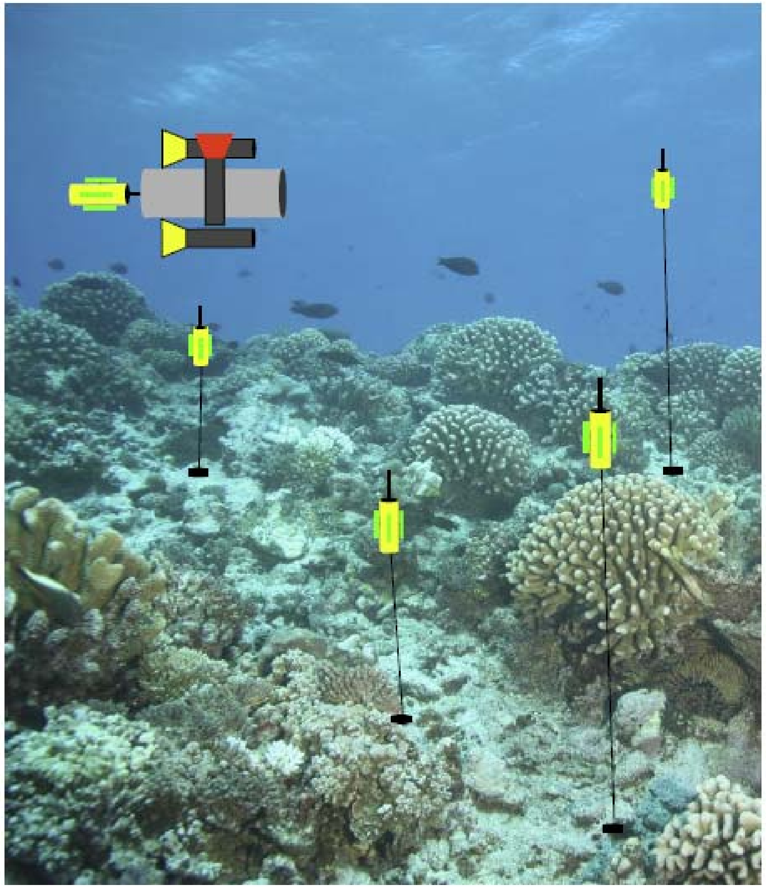
© Zlot, Stentz, Dias & Thayer



© Andrew Howard, USC/JPL

Monitoring Coral Reefs

- Automatically deploy sensors
  - To save energy
- Optimally place sensors
  - For improved sensing
- Communication between sensors and robot
  - For near real-time feedback



[From C. Detweiler, D. Rus et al, MIT]

Monitoring Coral Reefs – The AMOUR robot

- Easily deployed: 25.5 Kg 0.86m
- Shallow water operation (over 50m)
- 5 500W thrusters
  - 1.3 m/s
- >500Wh Li-Ion battery
  - 8km range
  - 6 hour typical operation time
- Vertical & horizontal orientation
- Adjustable buoyancy and balance
- On-board logging and sensing
  - Camera, depth, temperature, salinity, dissolved oxygen
- Acoustic, optical, & radio communication

[From C. Detweiler, D.Rus et al., MIT]

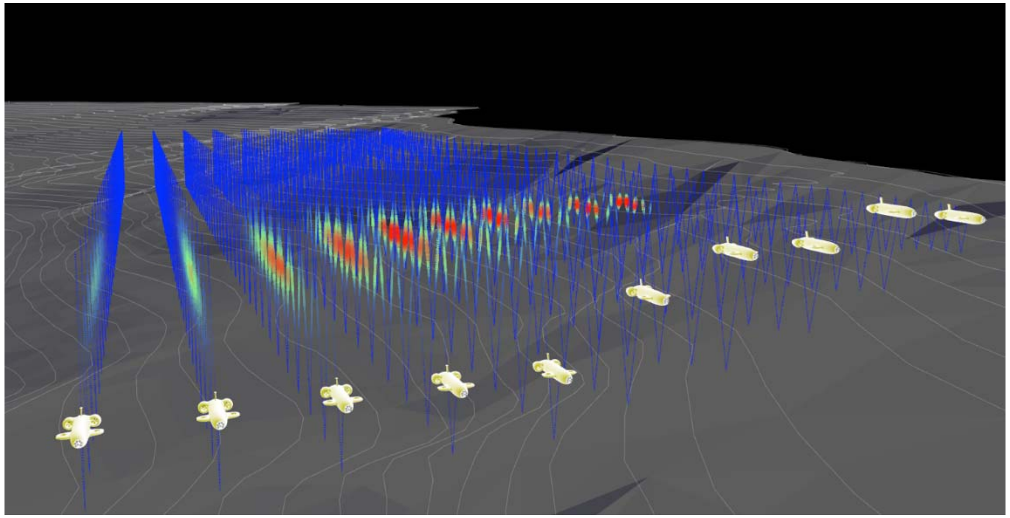
Monitoring Coral Reefs – TheAquaNodes

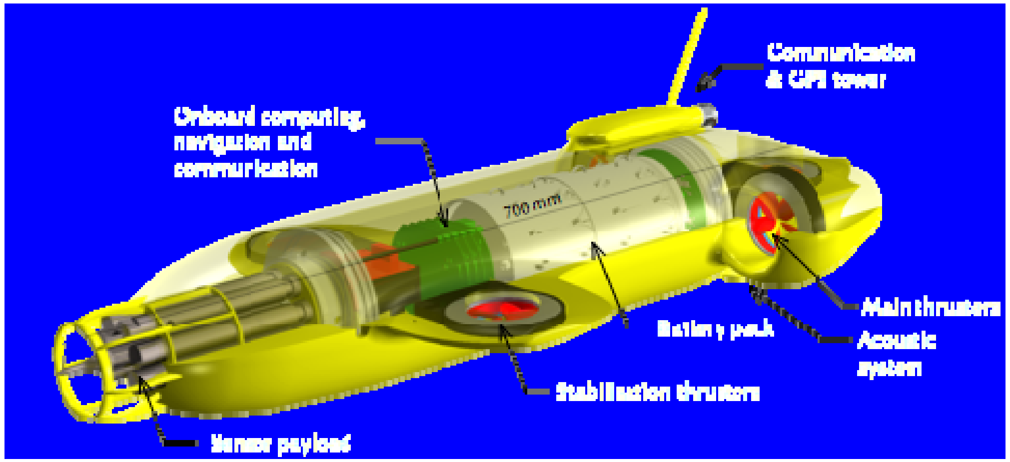
- Processor/Logging
  - LPC2148 60MHz ARM
  - SD Card
- Communications
  - Acoustic: 300b/s
  - Radio: 57kb/s on surface
  - Optical: 1Mb/s
- Sensors
  - Temperature, pressure, salinity, dissolved oxygen
  - Camera
  - Other digital and analog inputs
- Depth Adjustment
  - 0.5m/min

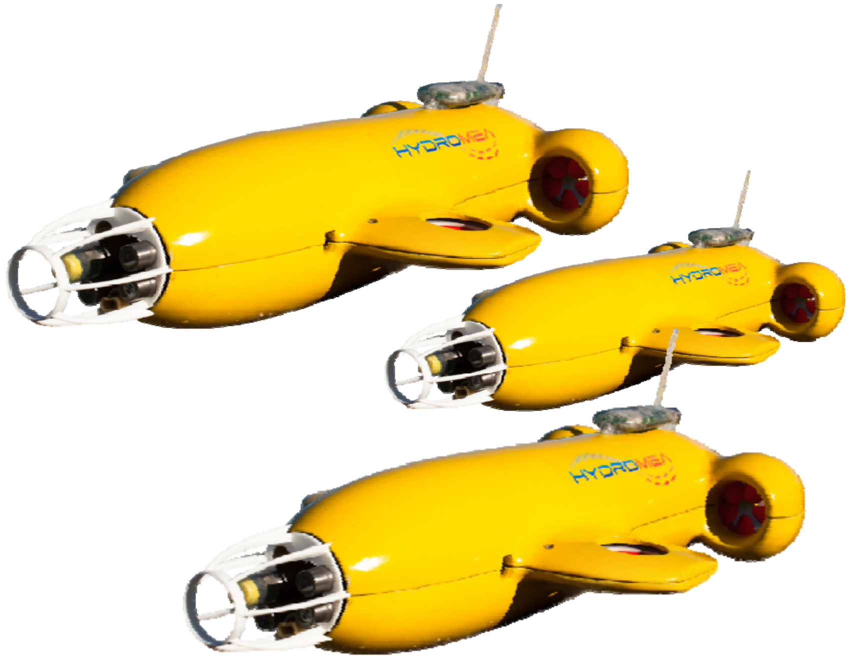
[From C. Detweiler, D.Rus et al., MIT]

Multi-AUVs for Limnology

[SNSF Sinergia project, 2015- Martinoli, Wueest, Ibelings; key personnel: Bahr, Schill]



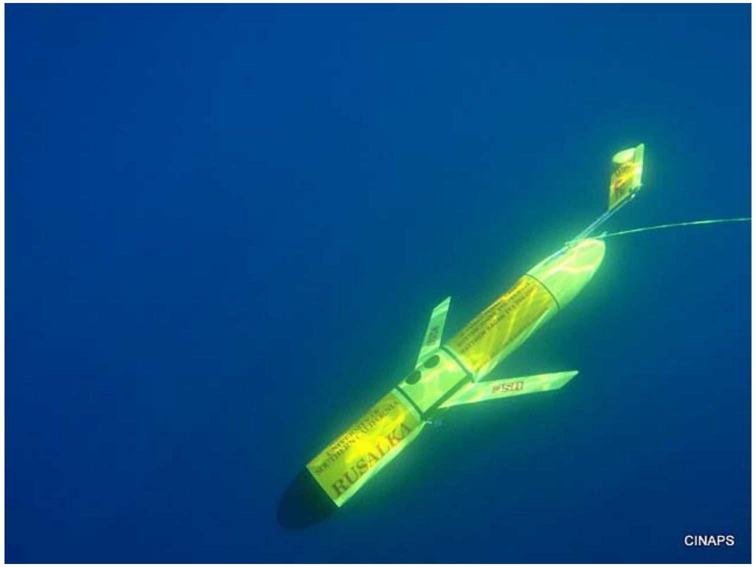




Buoys, Surface Vessels, and Gliders

- Water quality monitoring
- Alga bloom monitoring and early warning
- CINAPS: Center for Integrated Networked Aquatic Platforms

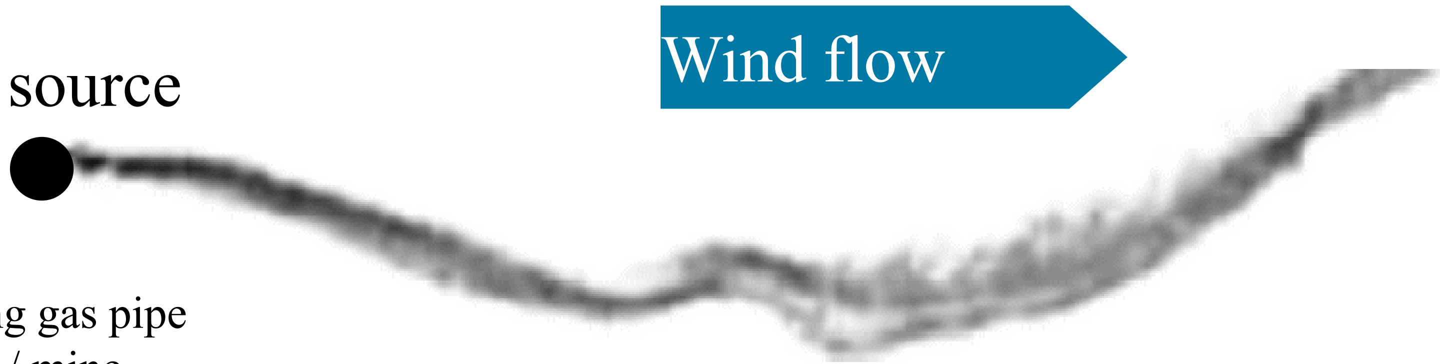
[G. Sukhatme et al., USC, CINAPS]



Robotic Sensor Networks for Odor Source Localization

Odor Source Localization

Odor source



e.g.:
leaking gas pipe
bomb / mine
food

Odor Source Localization

Wind flow

Odor source

e.g.:
leaking gas pipe
bomb / mine
food

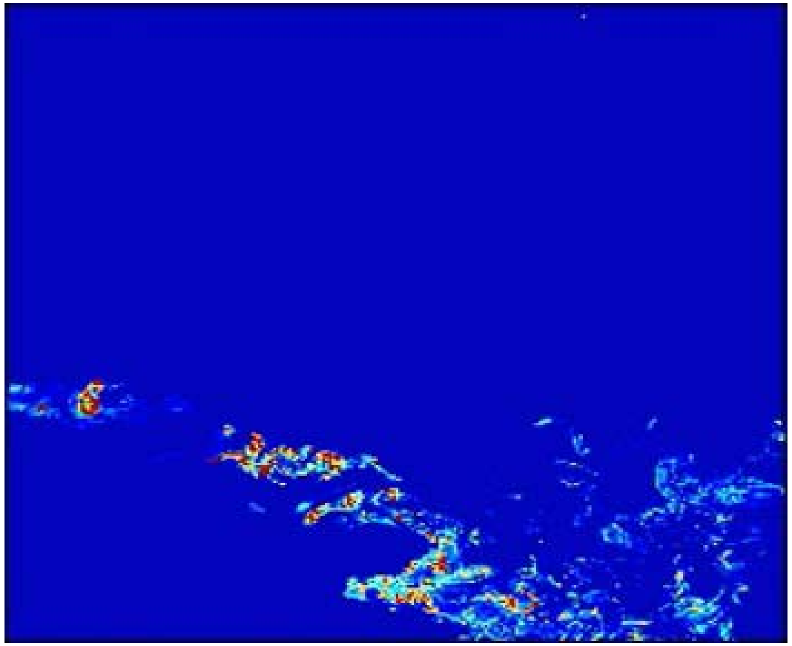
Khepera III robot with odor and wind direction sensor

Odor Source Localization

- Given:
  - Area to search (GPS bounds, enclosures, etc.)
  - Number of robots and hardware capabilities

- Three tasks (or phases though not always serial):
  - Plume finding
  - Plume traversing
  - Source declaration

Plumes: A Tricky Field to Traverse



Courtesy by L. Marques,
simulated plume

Algorithms

casting

surge-spiral

surge-cast

crosswind formation

probabilistic odor compass

probabilistic robotic search

computationally cheap — computationally expensive

dung beetle

function optimization (PSO, ...)

silkworm

biased random walk

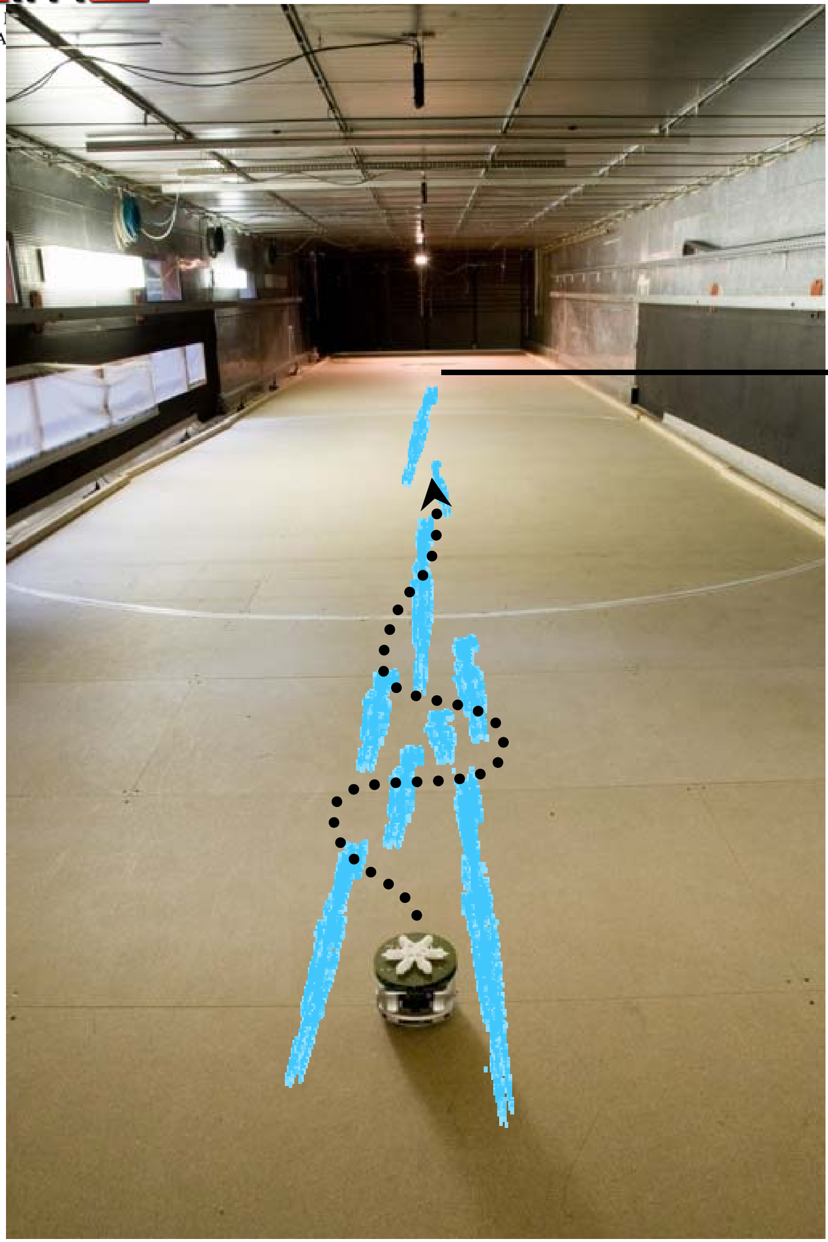
Braitenberg vehicle

infotaxis

probabilistic max. likelihood

Embedded Olfaction and Anemometry

- Khepera III robot (K-Team SA):
  - 13 cm diameter
  - differential-drive
  - 400 MHz ARM CPU
  - No floating point unit (FPU)
- Odor sensor board
  - VOC sensor, ppm level
  - Open sampling system
  - Actively sniffing (micro-pump)
- Wind direction sensor board
  - Thermistor-based array
  - Accuracy < 10 degrees



Systematic experiments in the wind tunnel

Arena: 18 x 4 m
Wind speed: 1 m/s (~laminar)

Single Robot Algorithms

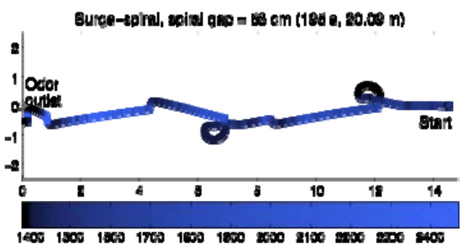
Bio-Inspired Algorithms

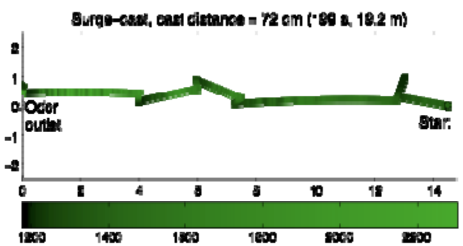
State machines, reactive

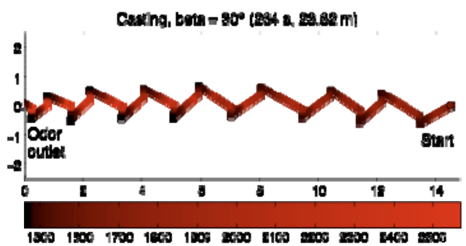
Combination of surging, casting, spiraling

Binary odor concentration
(in plume, not in plume)

Wind direction

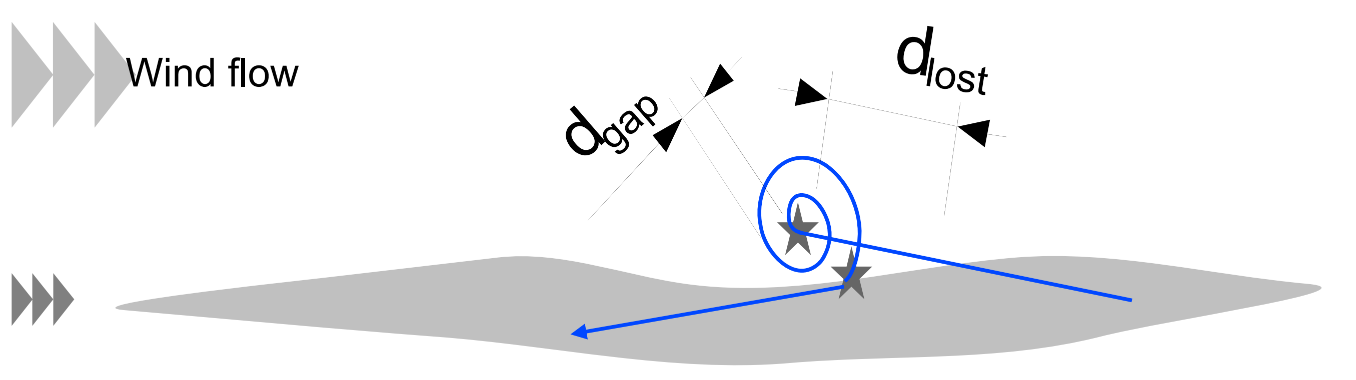






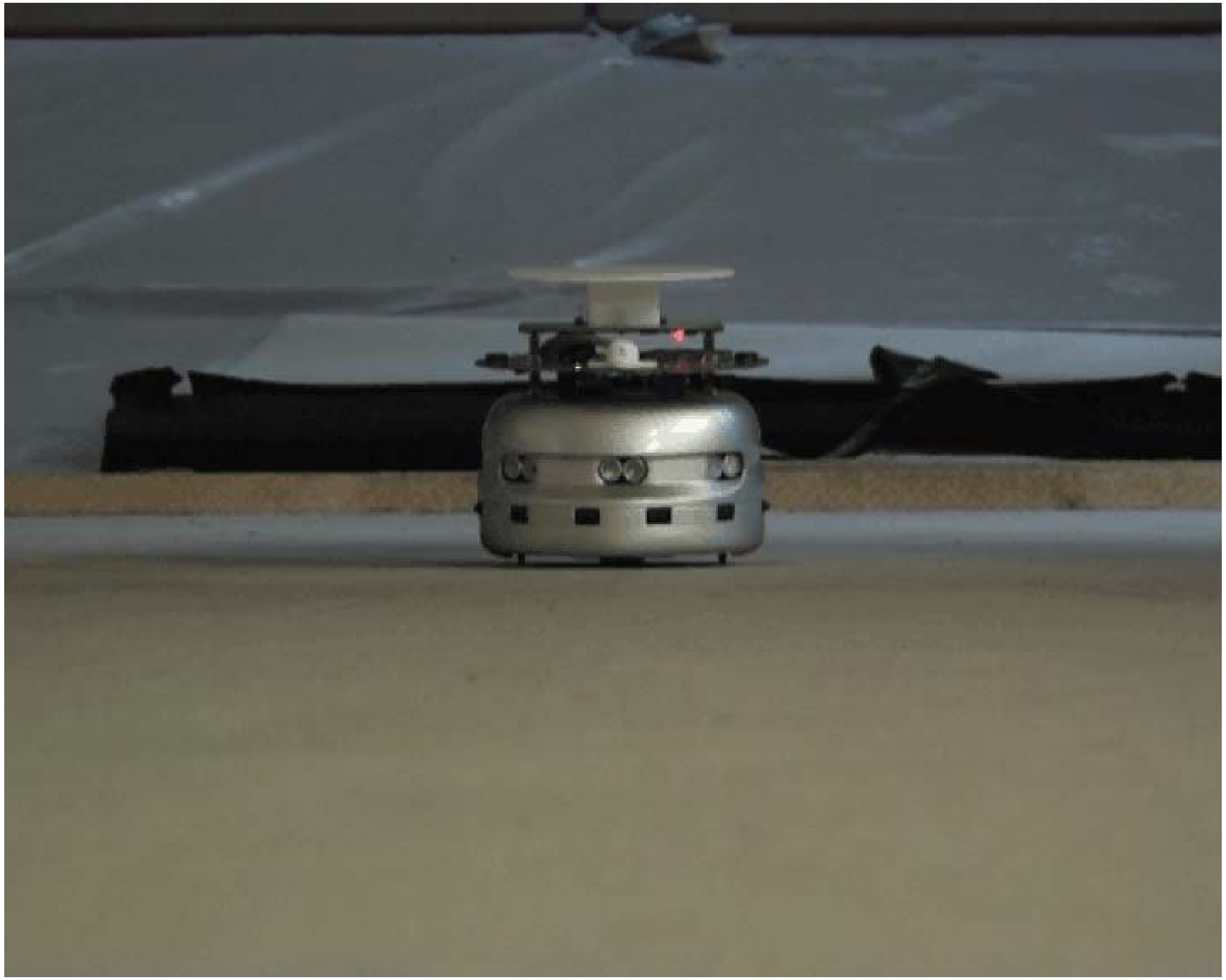
Surge-Spiral: Algorithm

Surge-spiral

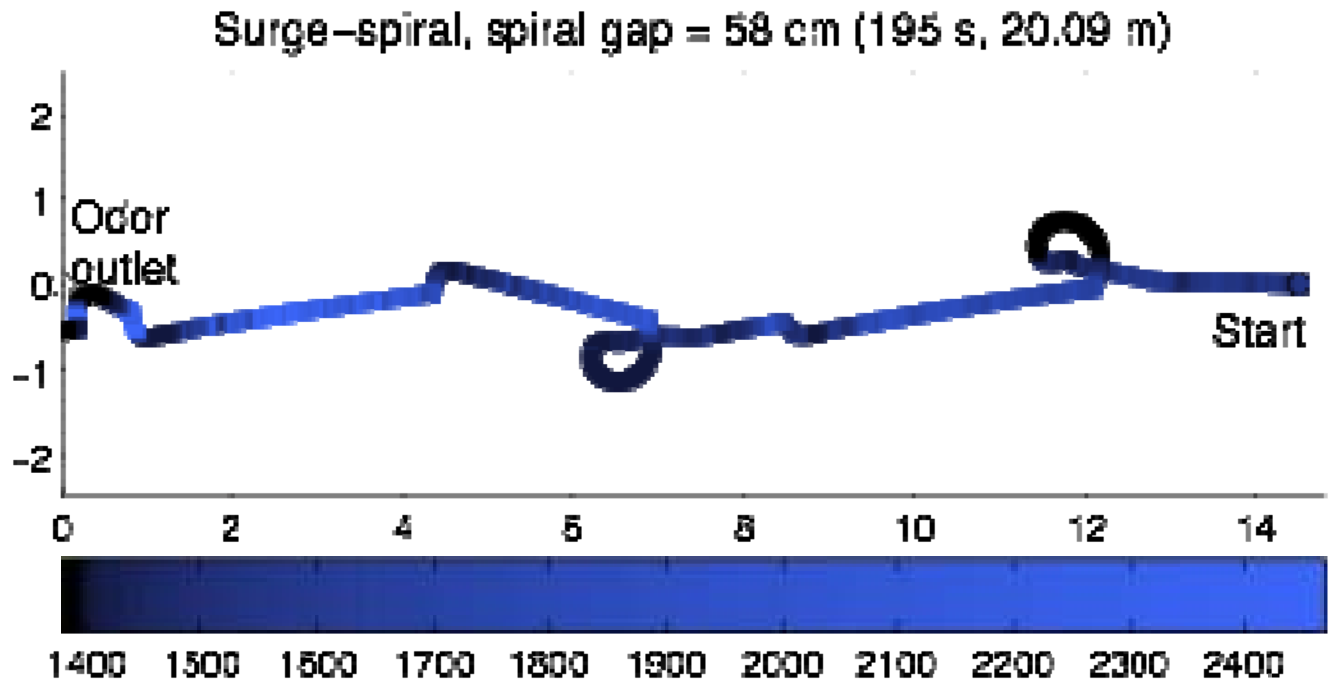


Wind direction measurement

Surge-Spiral: Real Robot Implementation

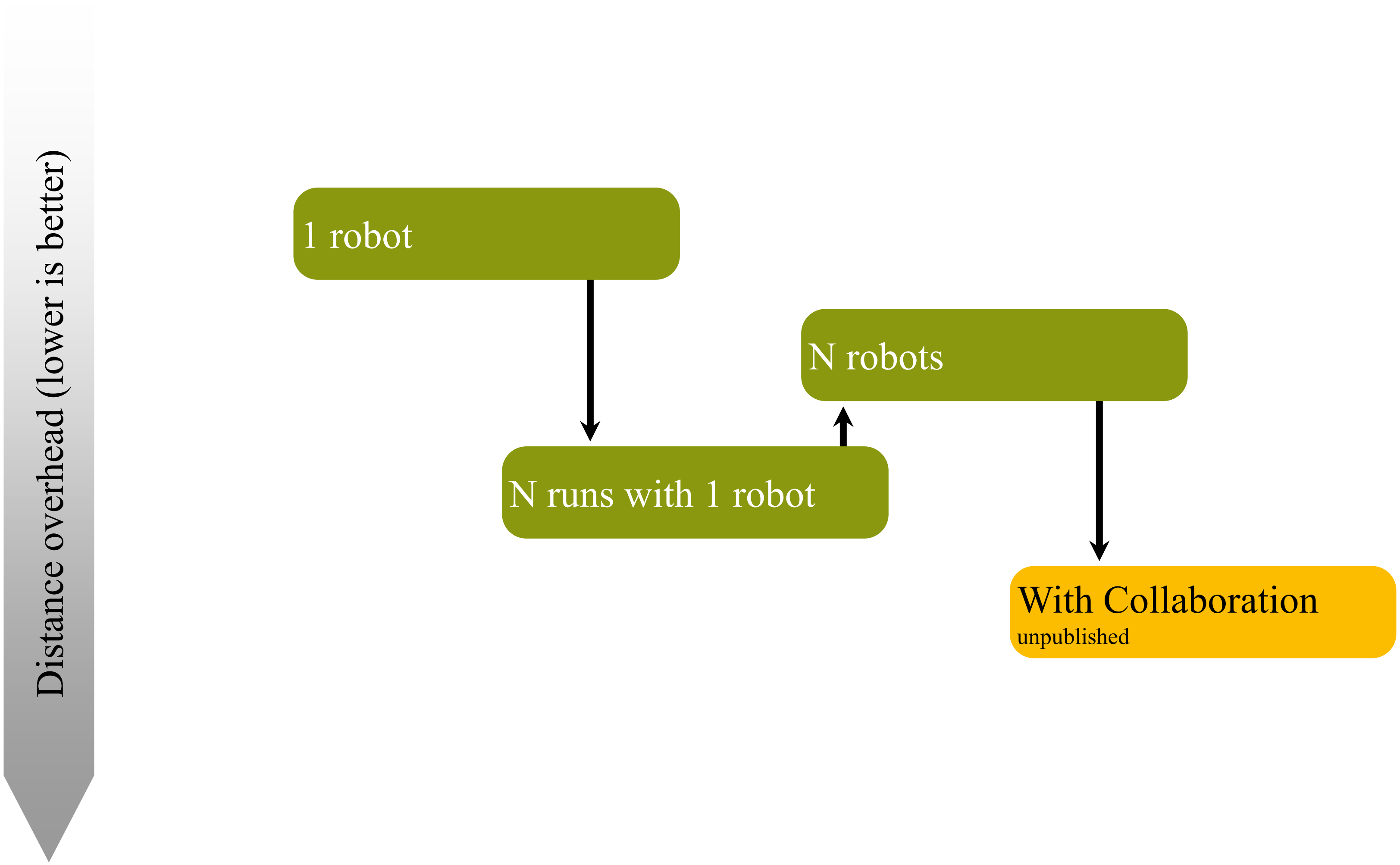


Surge-Spiral: Sample Trajectory



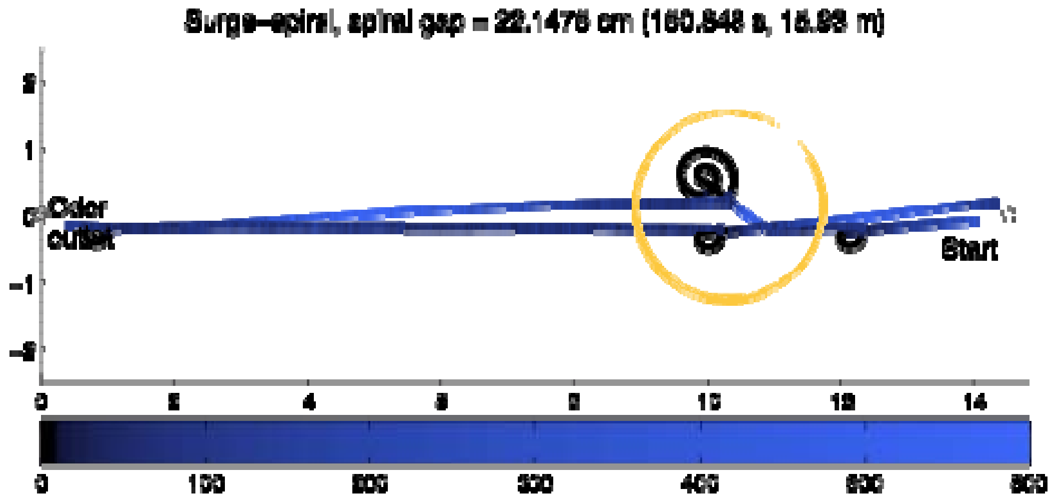
Upwind in the plume, spiraling for reacquisition

From Single Robot to Multi-Robot Algorithms



Distance overhead = D_{sf}/D_{min} normalized for single robot

Distance overhead (lower is better)



1 robot

N runs with 1 robot

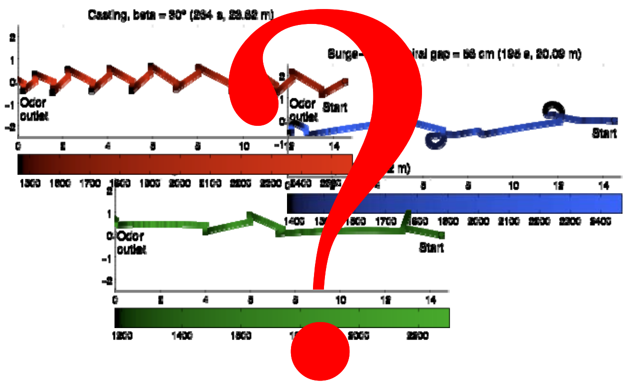
N robots

With Collaboration
unpublished

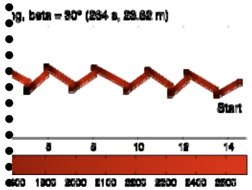
ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

With Collaboration

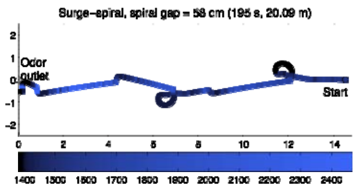
unpublished



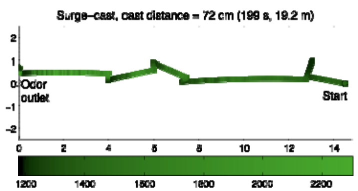
Improve?

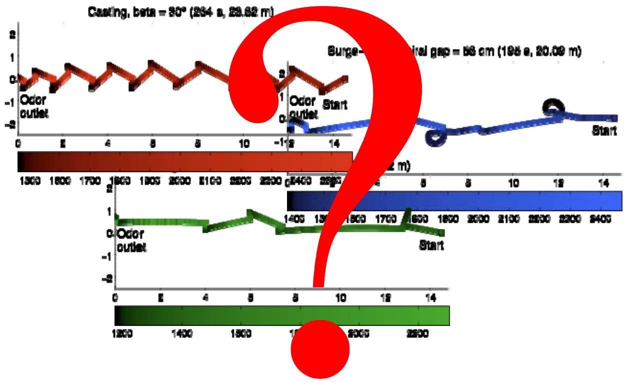


?



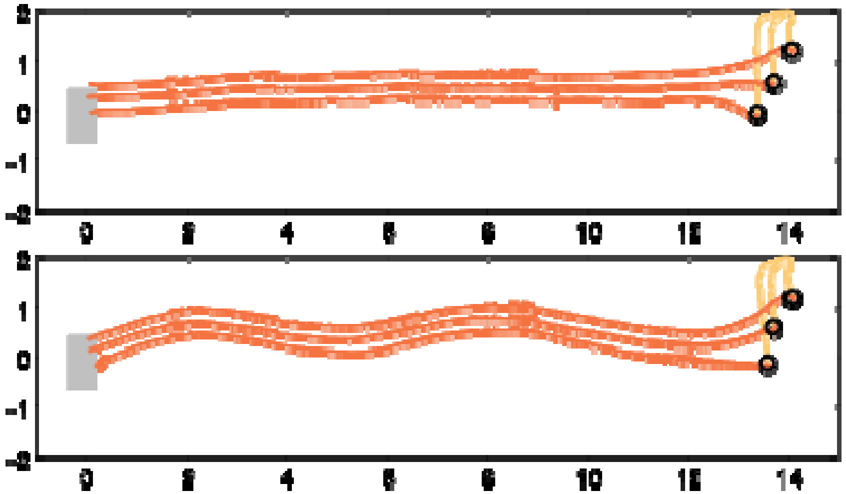
Redesign?





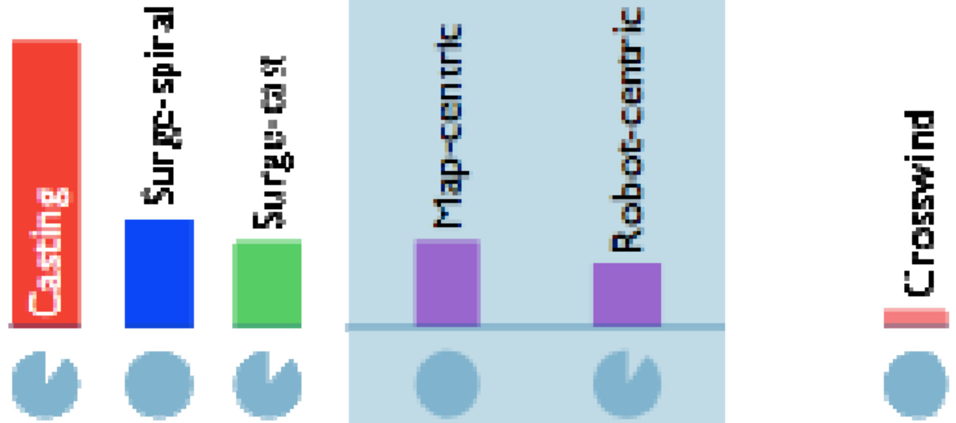
Improve?

Crosswind formation



Redesigned solution!

[Lochmatter et al, DARS 2010]

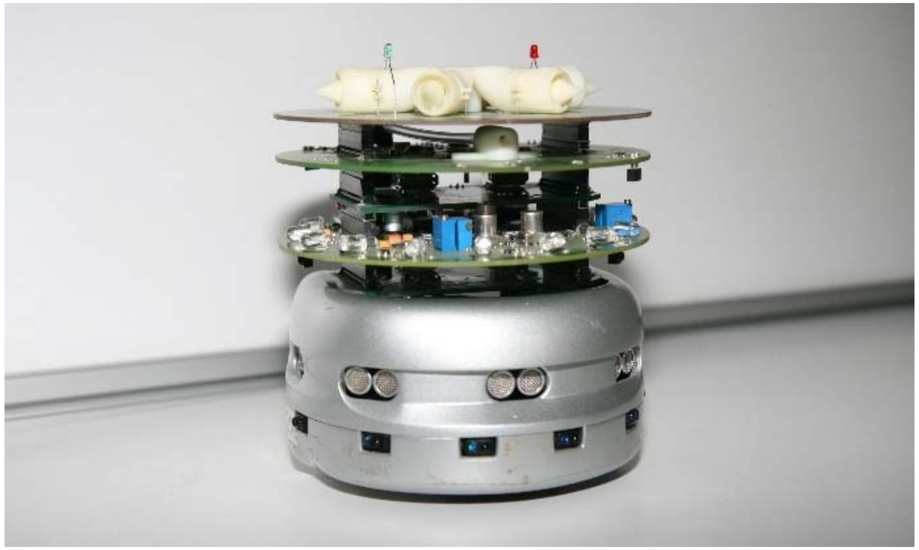


	bio-inspired	probabilistic		formations
	Casting, Surge-spiral, Surge-cast	Map-centric	Robot-centric	Crosswind
Distance overhead				
Success rate				
Single-robot	●	●	●	○
Multi-robot	○	●	○	●
Structure	state machine (bio. behavior)	discrete decisions (Bayes inference)		smooth adaptation (control theory)
CPU requirements (MIPS)	0.00	10'000	300	0.01
Memory requirements	few bytes	~ GB	< 50 kB	< 1 kB
Implementation	straightforward	hard	medium	straightforward
Plume model	implicit	explicit		implicit

Graph-Based Laplacian Control for Distributed Odor Source Localization

Approach and Robotic Node

- Odor concentration
- Wind direction
- Range and bearing
- Radio communication



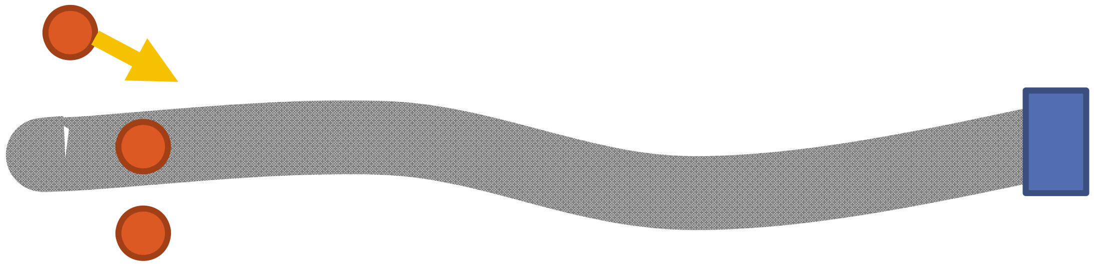
No global positioning

No external coordination

Behavioral Approach

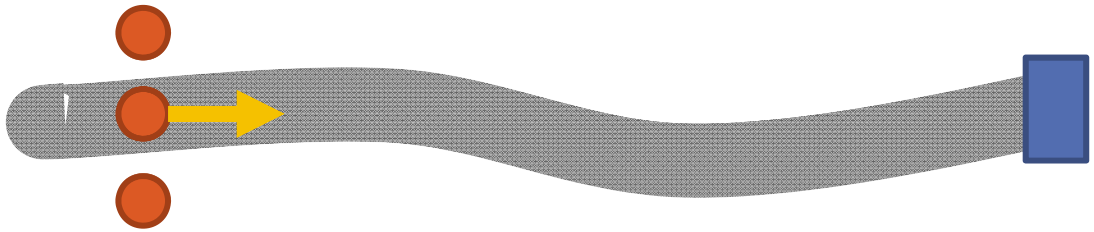
Formation control

Stay in formation



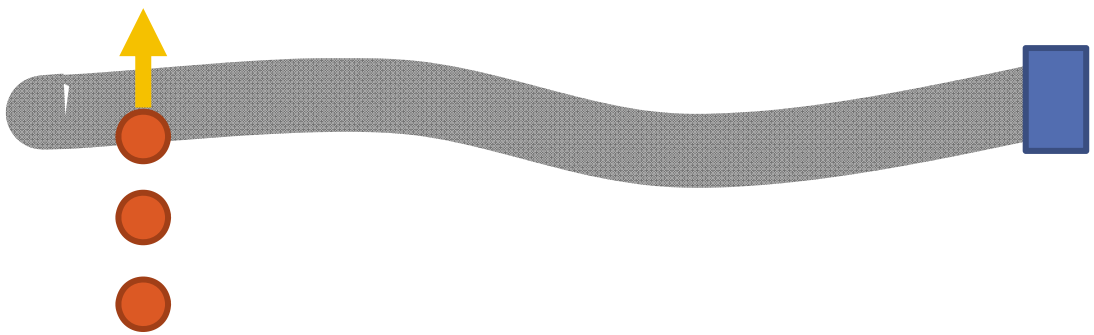
Upwind movement

Go towards the source



Plume centering

Stay in the plume



Laplacian Feedback Controller

Formation control

$$\mathbf{u} = -\begin{bmatrix} \sum_{j=0}^{N} \mathscr{L}_j \left(x_j - \beta_j^x \right) \\ \sum_{j=0}^{N} \mathscr{L}_j \left(y_j - \beta_j^y \right) \end{bmatrix} + R(\theta) \begin{bmatrix} 1 \\ -u_c^{max} + \frac{2u_c^{max}}{1+e^{-(c_l - c_r)/k_l}} \end{bmatrix}$$

Upwind movement + Plume centering

$$\beta_j = R(\theta) \begin{bmatrix} s_{uw} & 0 \\ 0 & s_{cw} \end{bmatrix} [\bar{\mathbf{p}}_\mathbf{j} - \bar{\mathbf{p}}_\mathbf{i}] \qquad s_{cw} = k_{cw} \frac{c_l + c_r}{1 + c_c}$$

$$v = k_v \mathbf{u}_x \qquad 0 \leq v \leq v_{max}$$

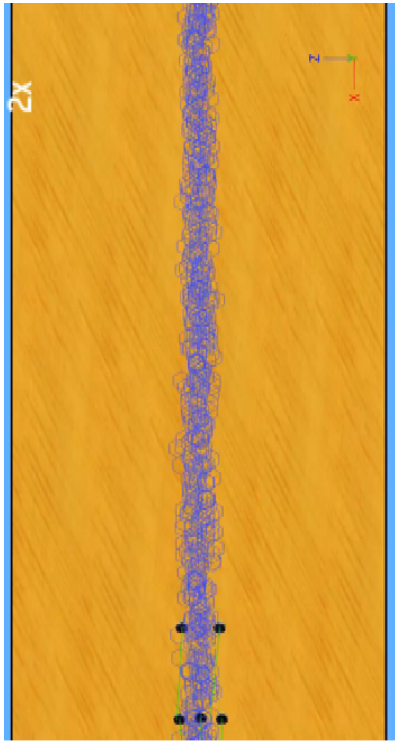
$$\omega = k_\omega \mathbf{u}_y \qquad -\omega_{max} \leq \omega \leq \omega_{max}$$

Θ = estimated wind direction (KF of wheel encoders and wind sensor board)
R(Θ) = rotational matrix
s_{uw} = scaling factor upwind direction; s_{cw} = scaling factor crosswind direction
c_c= center concentration; c_l = left concentration; c_r = right concentration
k_x = parameters (proportional gains)

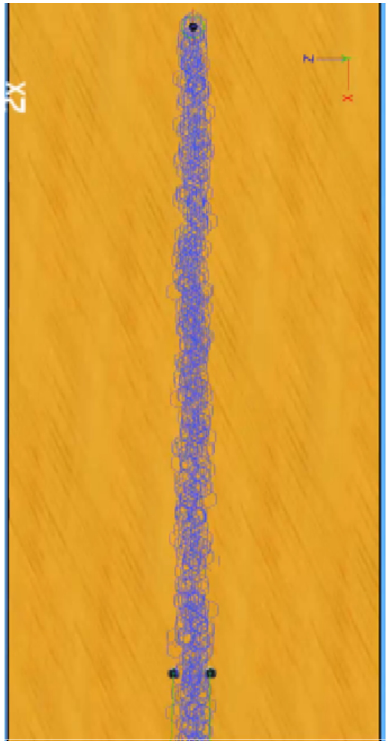
High-Fidelity Simulation

Formation control
Stay in formation

Upwind movement
Move towards the source

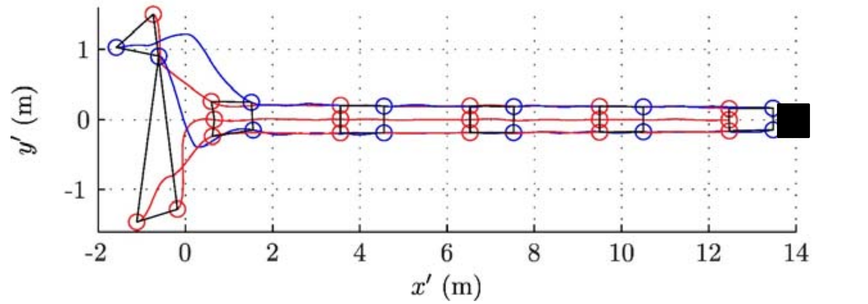


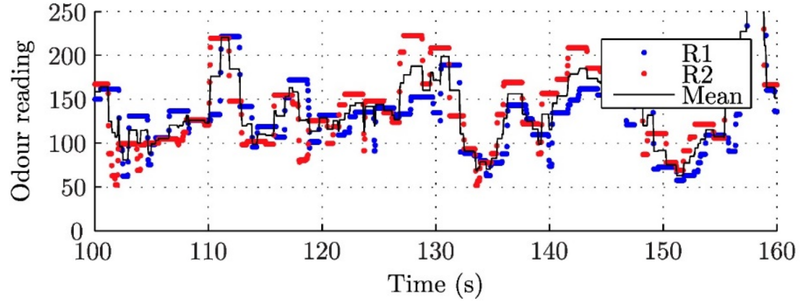
Plume centering
Stay in the plume

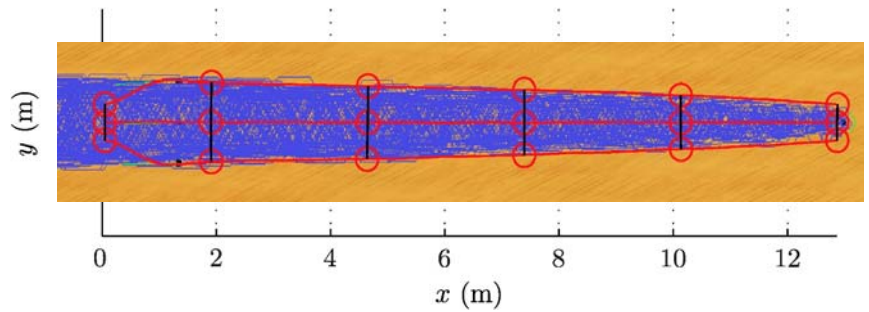


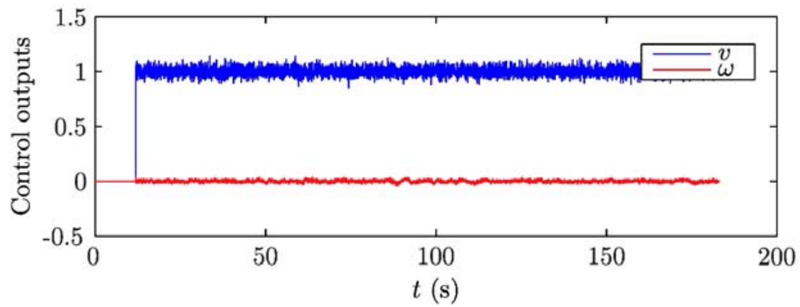
Webots + plume plug-in (filament model, [Farrell, 2002])

Trajectories and Odor Reading in Simulation







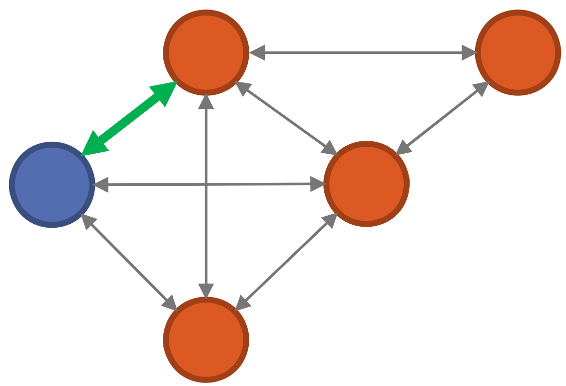


Algorithm Modifications for Wind Tunnel Experiments

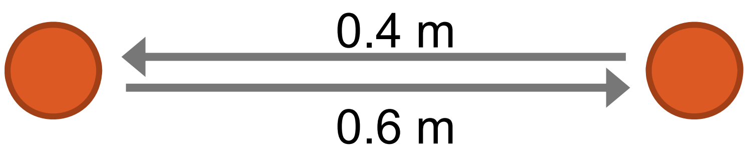
#1: SCALING FACTOR

$$\dot{s}_{cw} = k_{cw}((c_l + c_r) - c_c)$$

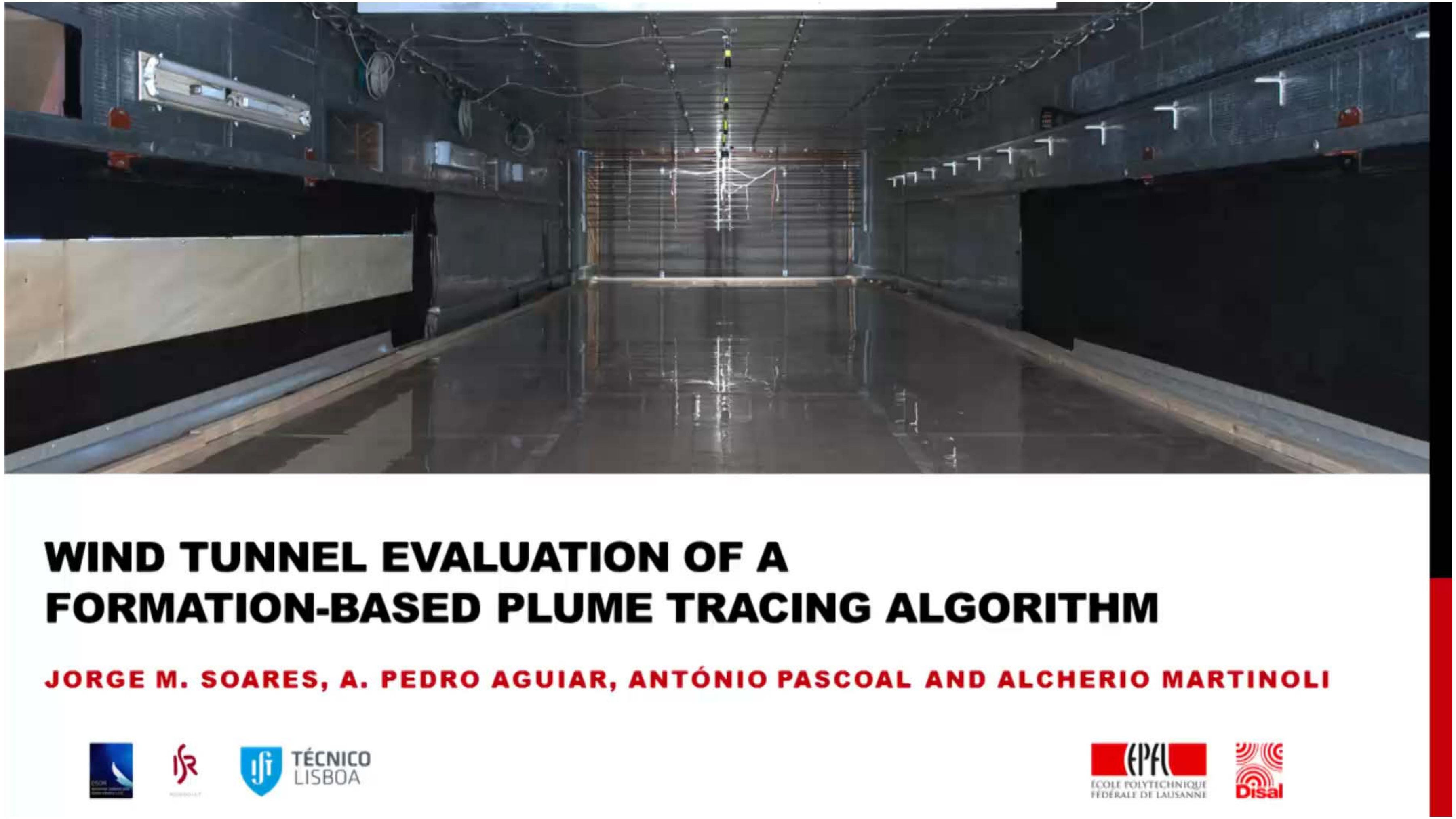
#2: COLLISION AVOIDANCE



#3: RANGE AVERAGING



Real Robots in the Wind Tunnel



[Soares et al., ICRA 2015]

Trajectories in the Wind Tunnel