ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Distributed Intelligent Systems – W9

Multi-Level Modeling: Calibration and Combination with Machine Learning

Outline

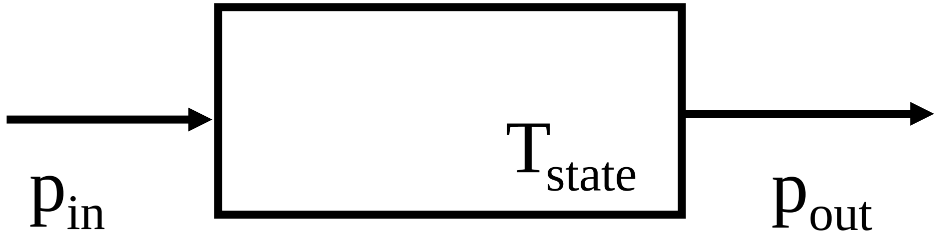
- Calibration methods for multi-level models
  - Microscopic and macroscopic parameters
  - Approximations
- A challenging example: distributed seed assembly
- Combined modeling and machine-learning methods
  - Homogenous and heterogeneous learning
  - Diversity and specialization

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Model Calibration

From W8: Parameter Calibration

- Number of parameters is decreasing with the abstraction level
- Calibrate a given level based on the underlying one (e.g., submicroscopic with physical system; microscopic with submicroscopic, macroscopic with microscopic)
- Parametric (e.g., mean only, mean and variance) or non parametric (actual distribution recorded at the lower level) assumptions
- Various methods available
  - Ad hoc experiments [Correll & Martinoli, ISER 2004]
  - System identification techniques (e.g., constrained parameter fitting) [Correll & Martinoli, DARS 2006]
  - Statistical verification techniques (e.g., trajectory analysis) [Roduit et al., IROS 2007]
- Parameter example for micro- and macroscopic models:
  - State durations
  - State transition probabilities



From W8: State Durations & Discretization Interval

1. Measure all interaction times of interest in your system, i.e. those which might influence the system performance metrics.

   Note: often "delay states" can just summarize all what you need without getting into the details of what's going on within the state.

2. Consider only average values (we might consider also parameter distributions in the future, the modeling methodology does not prevent to do so)

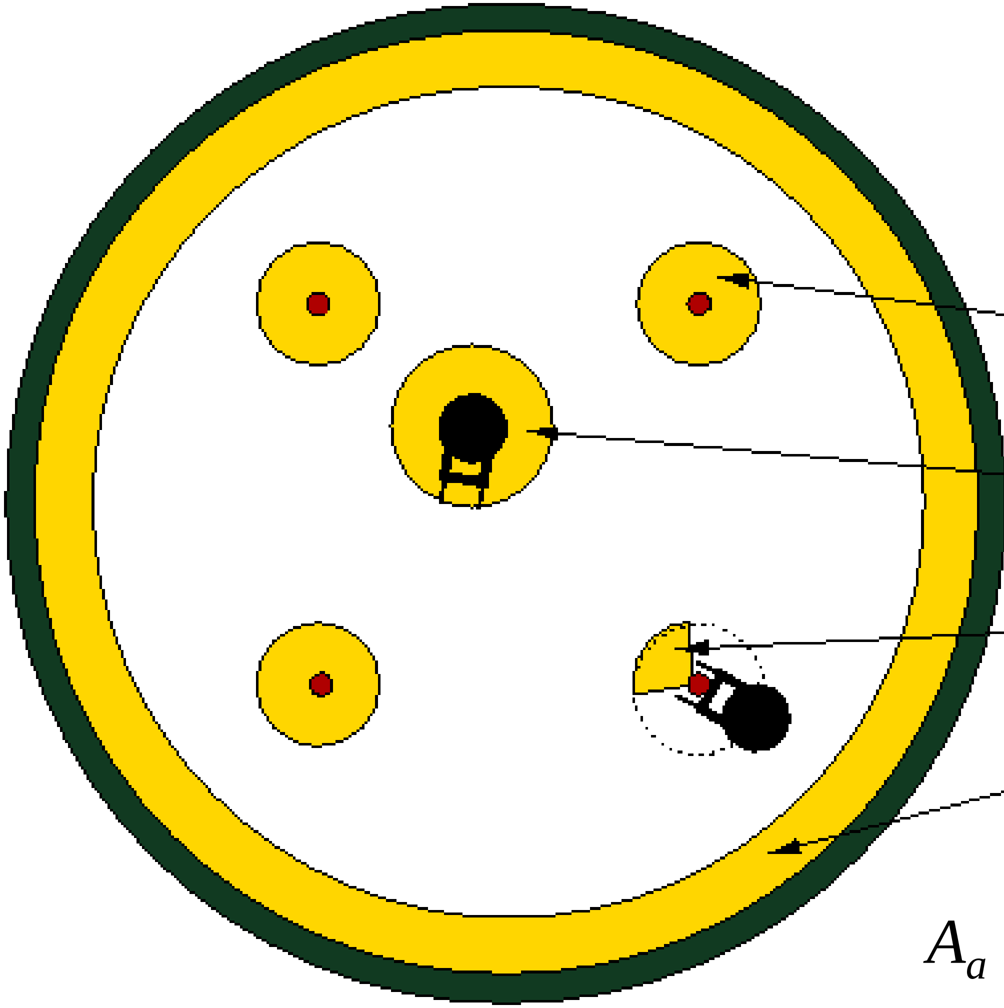
3. For time-discrete systems: choose the **time step T** = GCF of all the durations measured (e.g., 3 s obstacle avoidance, 4 s object manipulation, T = 1 s) -> no rounding error.

   Note: more accuracy in parameter measuring means in this case more computational cost when simulating

From W8: State Transition Probabilities

Note: As defined in [Martinoli et al, IJRR 2004]!

- Geometric considerations
- Ad hoc calibration experiments
- Ex. stick-pulling experiment



$$p_s = A_s / A_a$$

$$p_r = A_r / A_a$$

$$p_R = p_r(N_0 - 1)$$

$$p_w = A_w / A_a$$

$$p_{g1} = p_s$$

$$p_{g2} = R_g p_s$$

A_a = *surface of the whole arena*

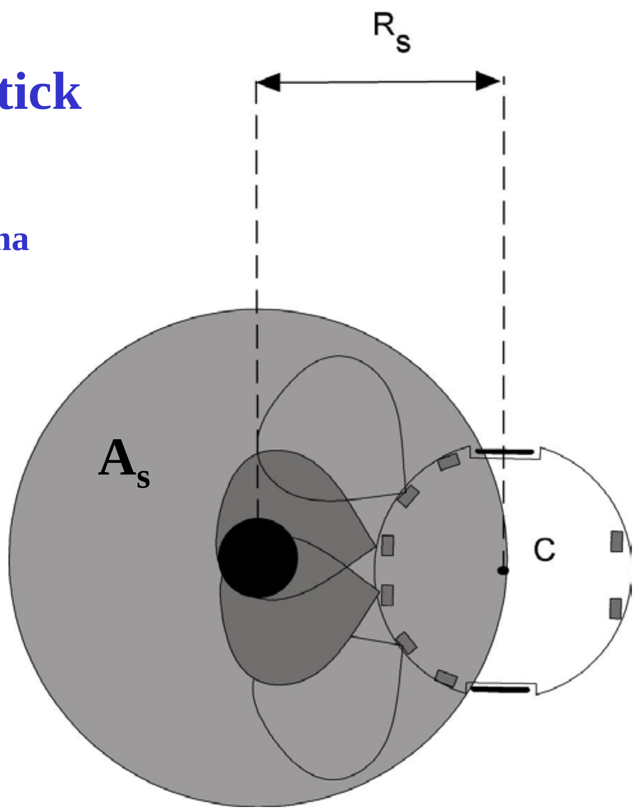
Geometric Probabilities g_i

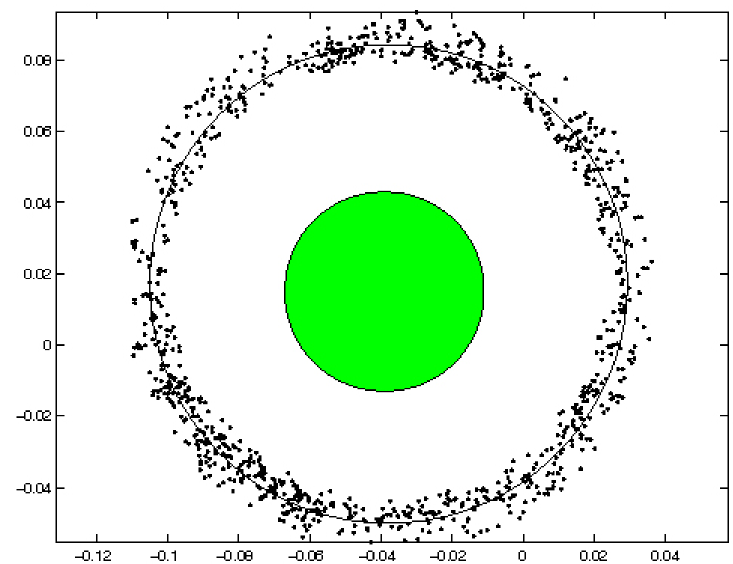
[Correll & Martinoli, ISER 2004]

- g_s, g_w, …are function of sensor range, behavior, robot's and object's size, … : interaction characterization!
- Geometric probabilities can be considered normalized detection areas (normalized over the total area of the experiment).

Example: stick

$g_s = A_s/A_{arena}$





ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Encountering Probabilities

[Correll & Martinoli, ISER 2004]

1. Measure geometric probabilities of detection g_i
2. Calculate the **encountering rate r_i** [s^{-1}] for the object **i** from the **geometric probabilities g_i** :

$$r_i = \frac{vW_s}{A_s} g_i$$

A_s = detection area of the smallest object

v = mean robot speed

W_s = robot's detection width for the smallest object (center-to-center)

3. For time-discrete models, calculate the **encountering probabilities p_i** (per time step) from the encountering rates:

$$p_i = r_i T$$

Note: slightly different from [Martinoli et al., IJRR04] (decoupled time and space) !

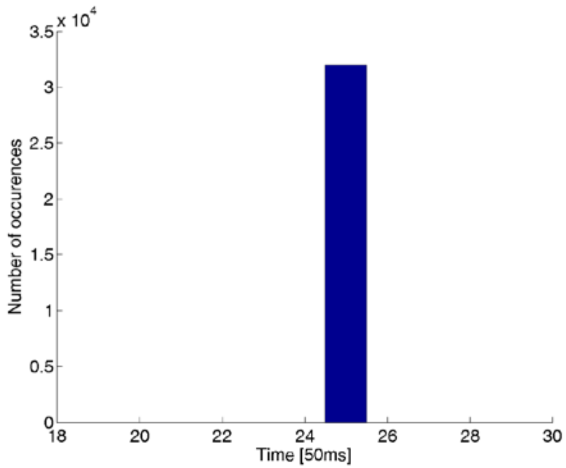
Model Calibration - Practice

- Assumptions (well-mixed, linear overlap of areas) might be only partially fulfilled
- We do not capture distributions in the model parameters, only deterministic average values; distributions might more faithfully capture:
  - Controller type (e.g., distal vs. proximal)
  - Active vs. passive objects (e.g., robot vs. wall)
  - Embodiment vs. non embodiment (e.g., area vs. real obstacle)
  - Way of measuring your metrics (e.g., egocentric, allocentric)
  - Impact on the considered swarm performance metric through error propagation (clear decoupling between parameters and structure inaccuracies of the model)

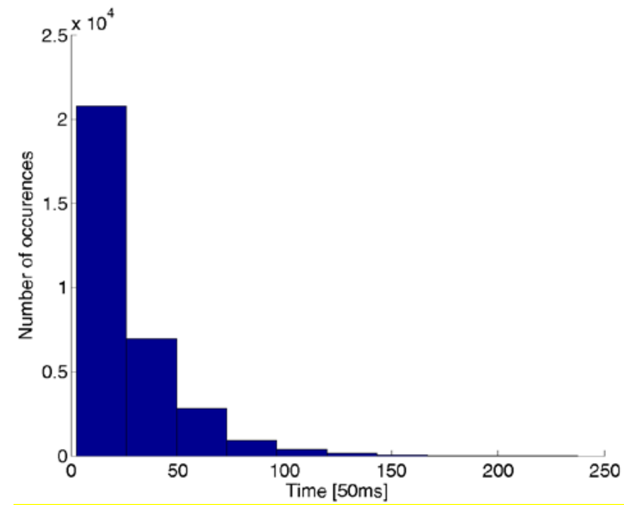
Model Calibration - Practice

Bin distribution of interaction time T_a (mean T_a= 25 *50 ms = 1.25 s)

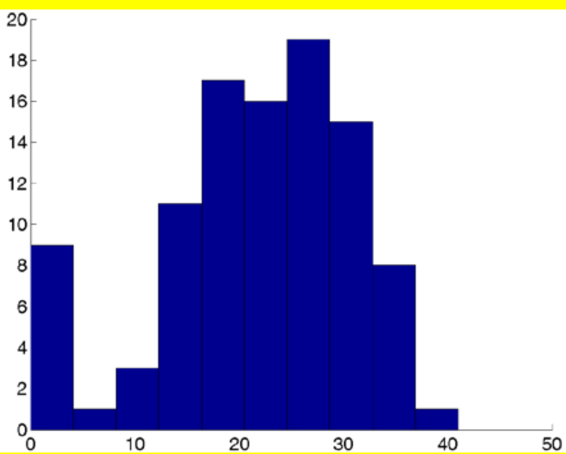
# of collisions



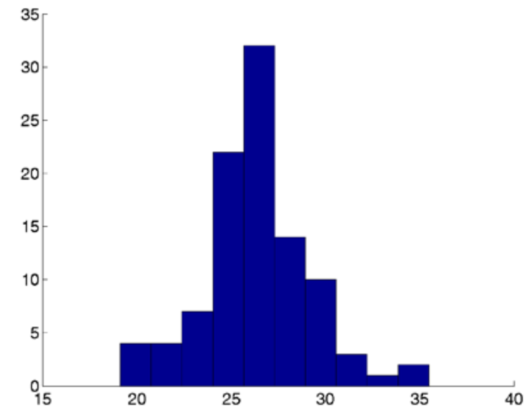
Micro/macro, deterministic delay



Micro/macro, prob. delay



Sub-microscopic, distal controller

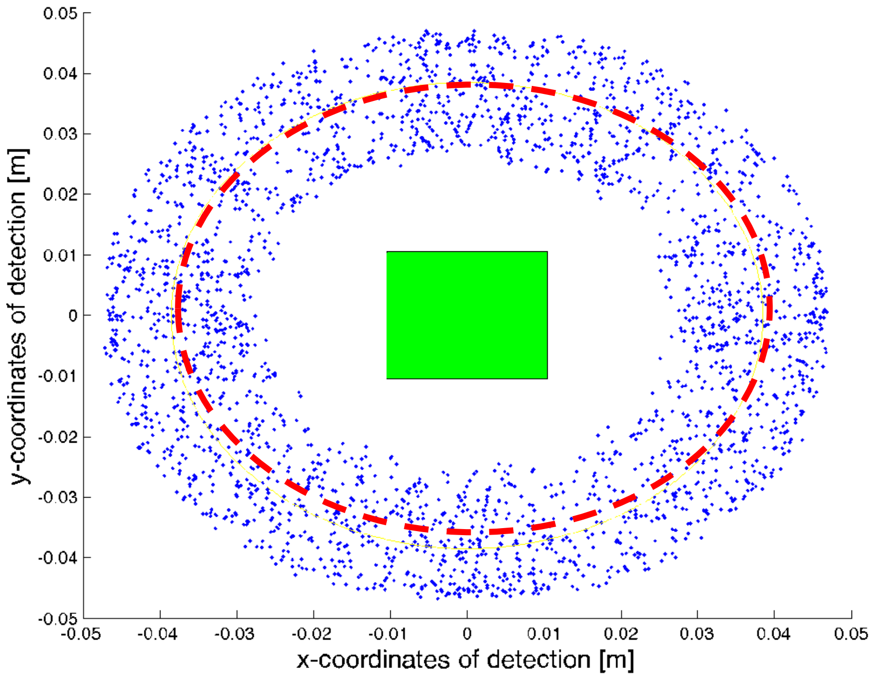


Submicroscopic, proximal contr.

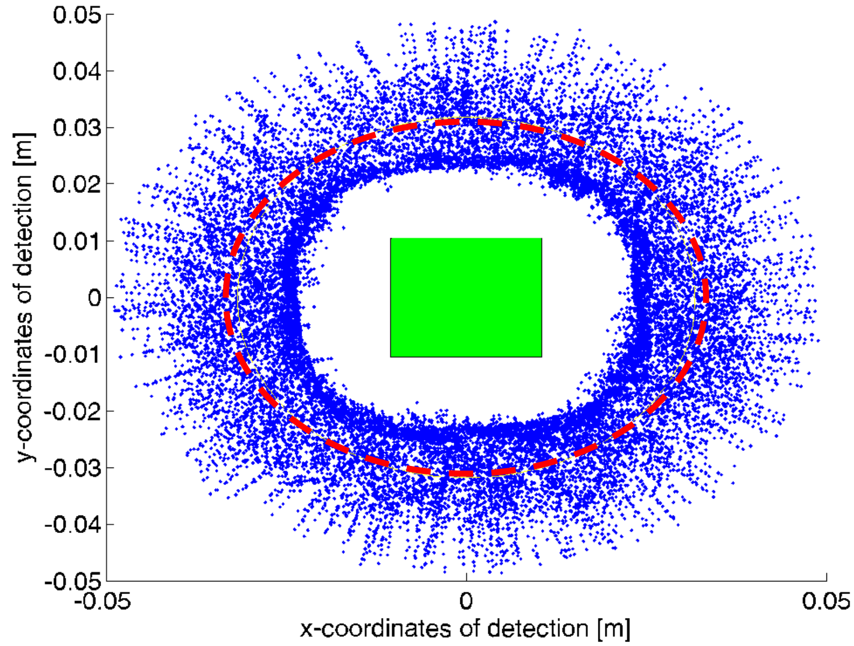
Collision time

Model Calibration - Practice

Geometric probability g: example of transition in space from search to obstacle avoidance (1 moving robot, 1 dummy robot, Webots measurements, egocentric)



Distal controller
(rule-based)



Proximal controller
(Braitenberg, linear)

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

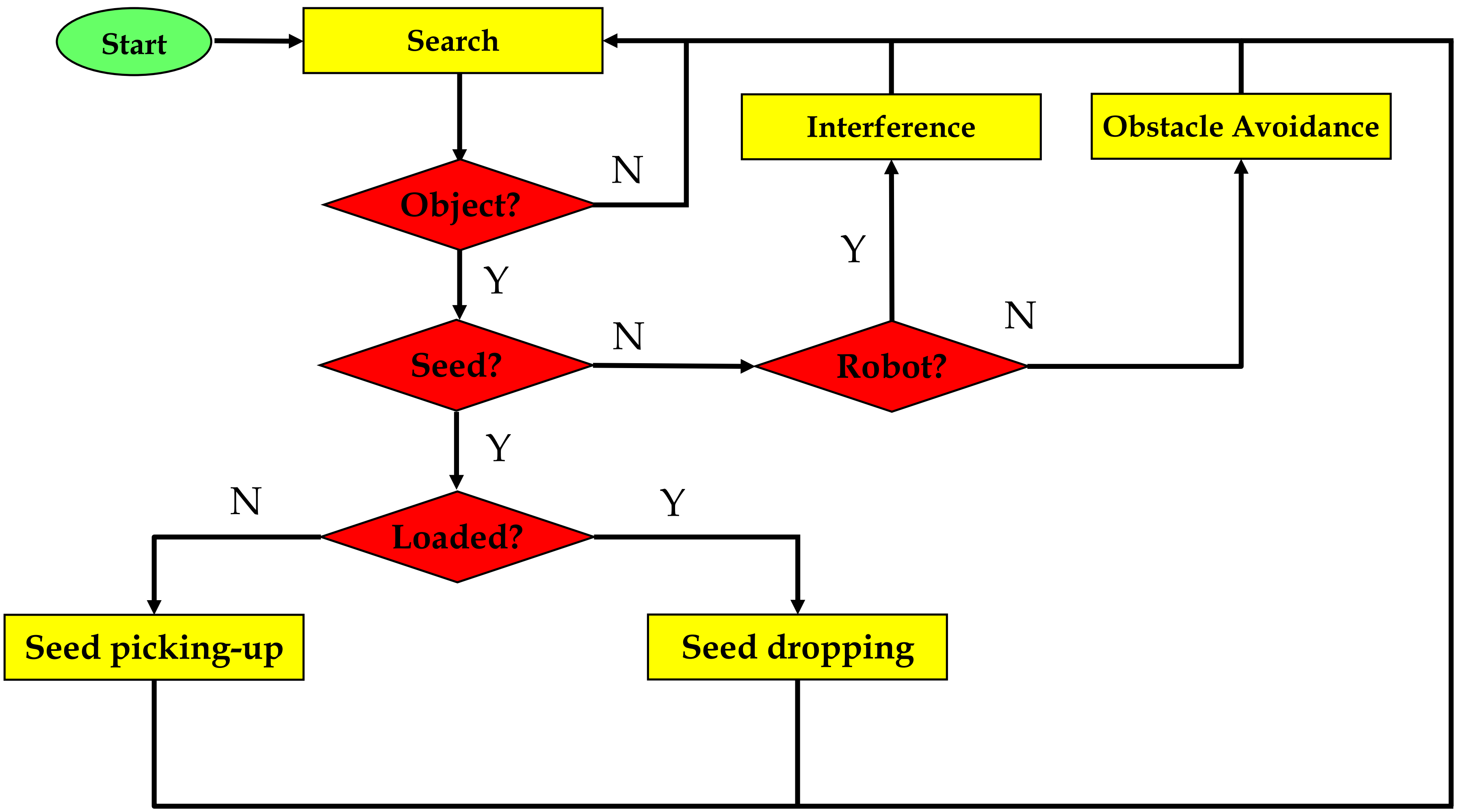
A Challenging Example: Distributed Seed Assembly

The Seed-Assembling Case Study

Robot behavior

- Reactive, non- communicating, non-adaptive behavior
- Qualitative stigmergy important: 2 rules in interaction with cluster:
  - Avoid if interaction with the cluster body
  - Manipulate if interaction with cluster tips
- Quantitative stigmergy minimal :
  - the bigger, the more stable the cluster
  - big cluster (> 2) = number of manipulation sites as cluster of 2 seeds
  - almost no difference between cluster incrementing and decrementing probabilities
- 1 robot state: loaded, free

Robot Controller

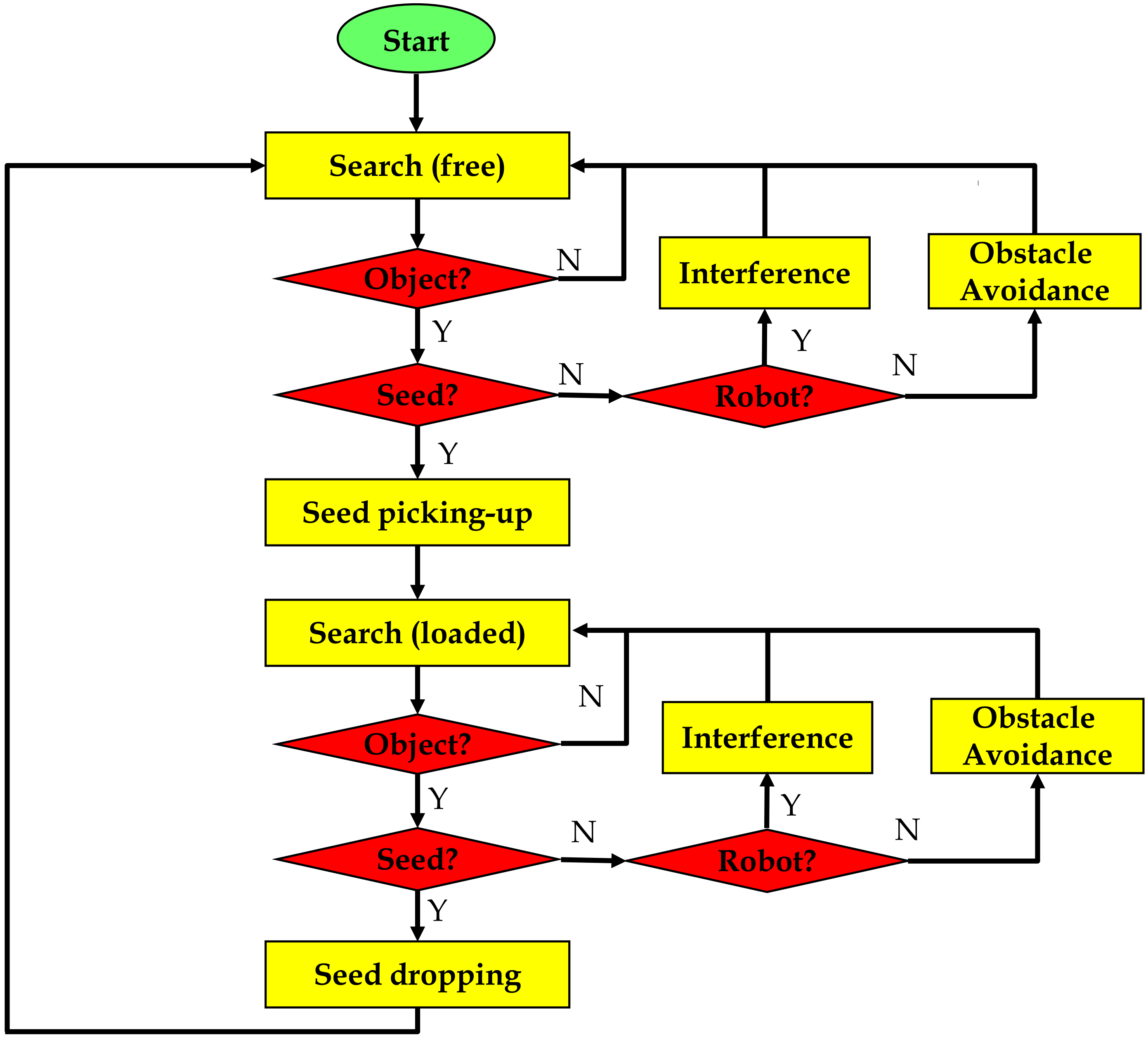


State Granularity Choices

- Idea: split the search state in search (loaded) and search (free) and keep dedicated avoidance/interferences states for each of the split states

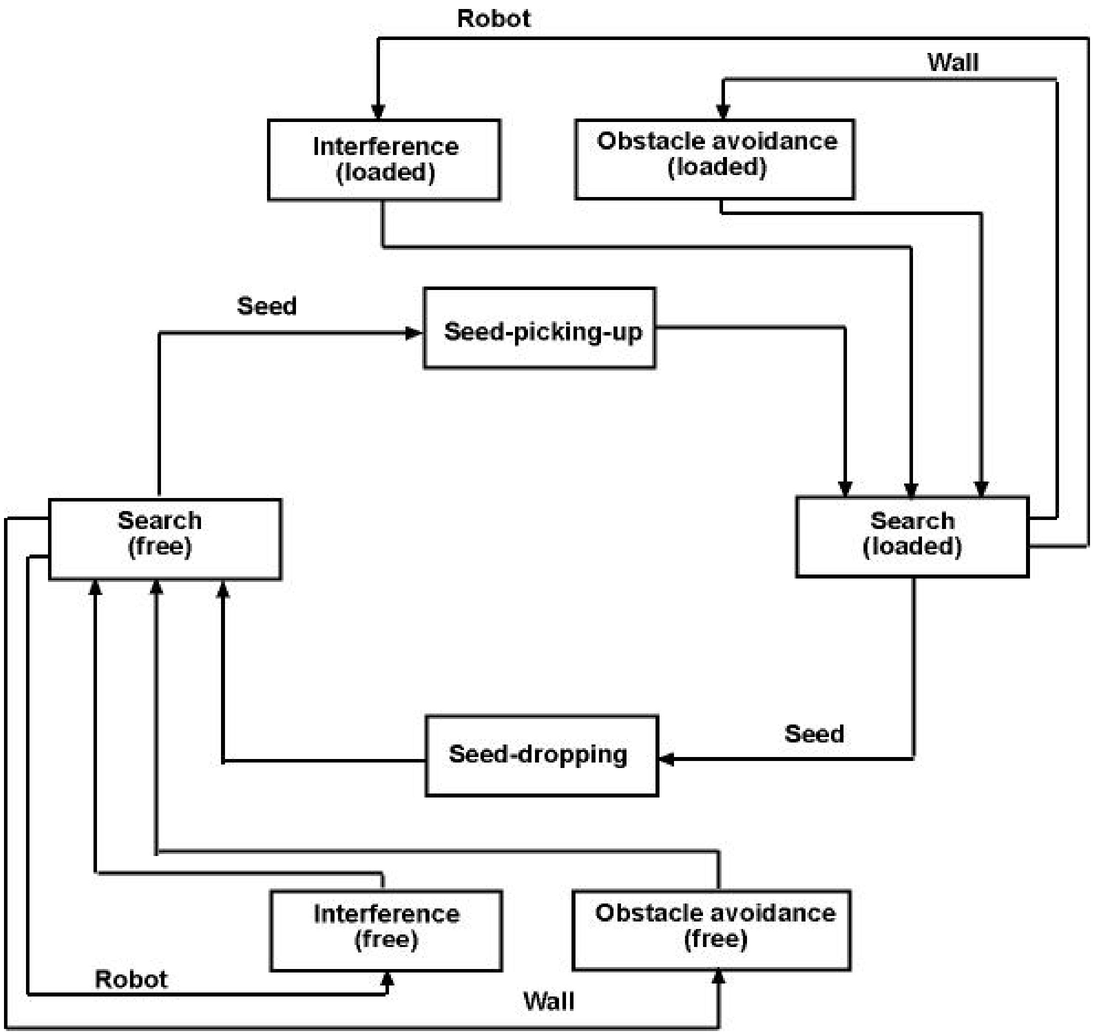
- Motivation:
  - Chosen metrics: assembly progress based on seeds on ground → seeds need to be tracked also when they are in the robot grippers (conservation law of seeds)
  - Status loaded (carrying a seed) or free (not carrying a seed) can only change in a deterministic fashion (e.g., cannot be changed by an avoidance operation) and only by going to one of the seed dropping or picking states → can be explicitly represented with a larger state space and deterministic transitions
  - Such enlarged state space facilitates the writing of the ODEs

Robot Controller – Restructured

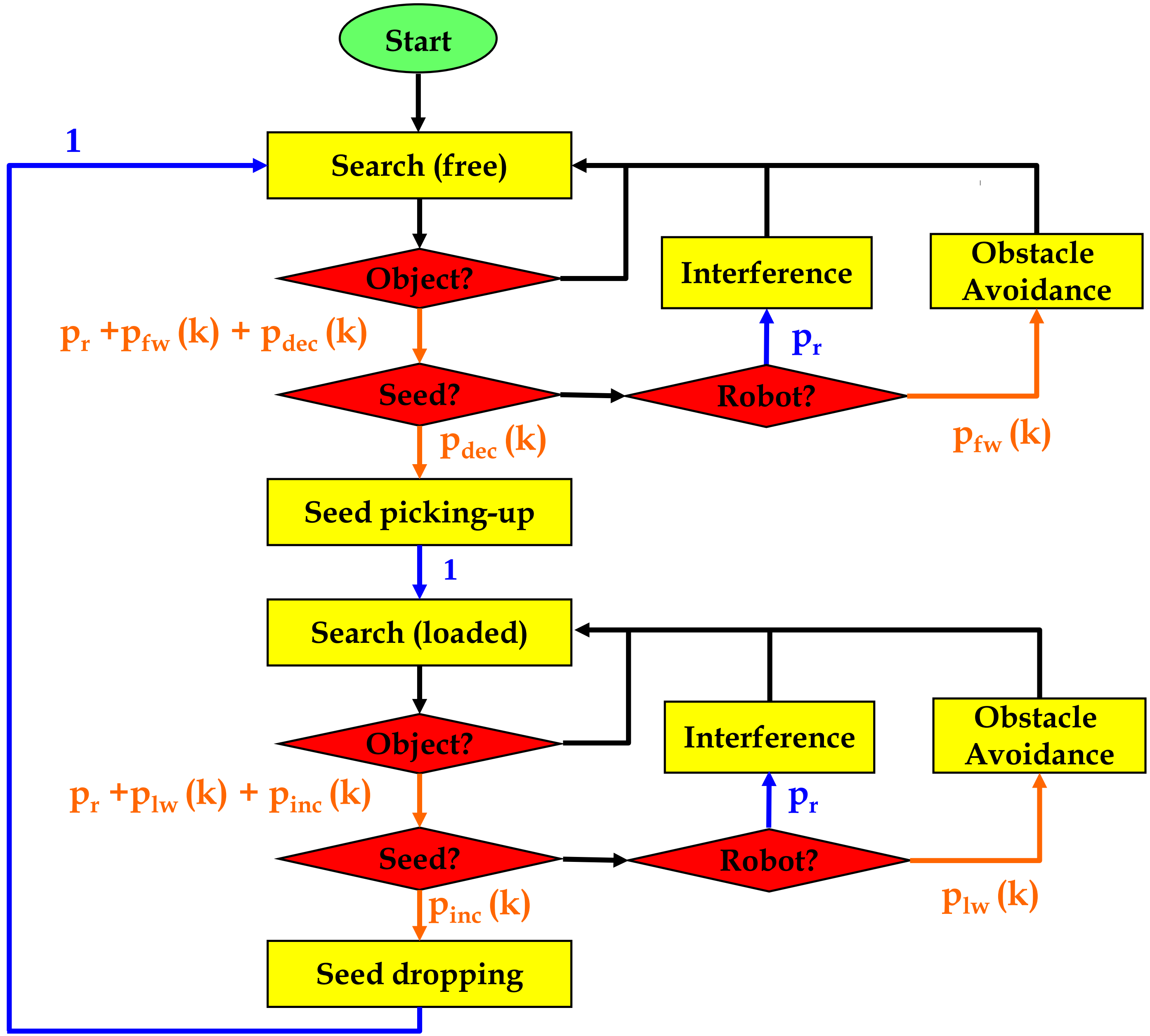


Robot Controller

From Agassounon et al, 2004 (restructured representation):



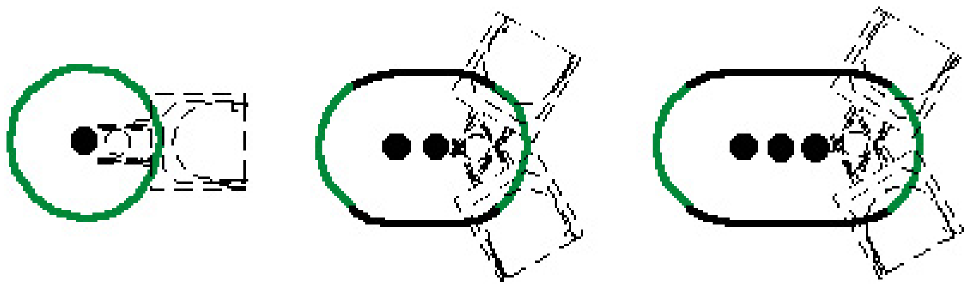
Robot Controller – Restructured



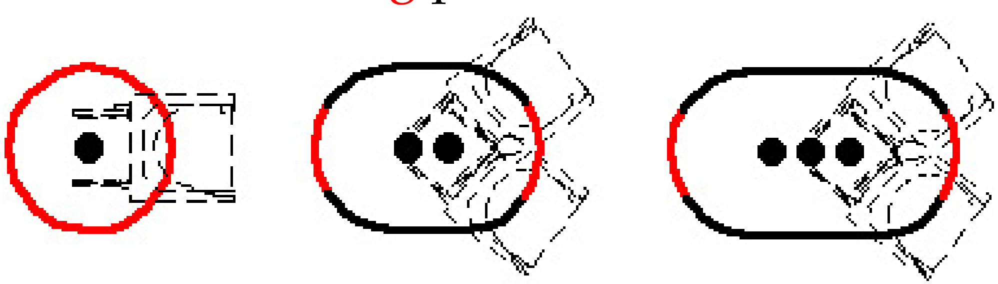
Parameter Calibration

Geometric Estimations

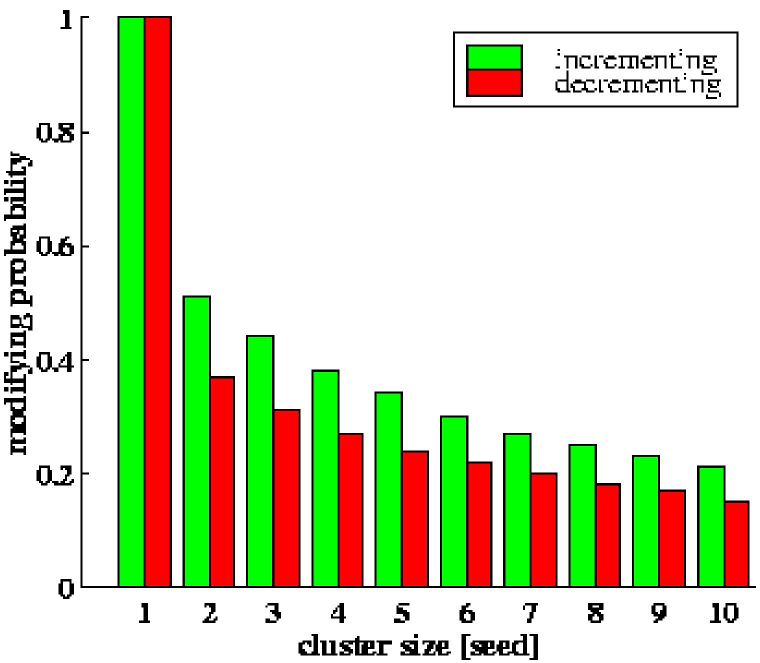
- Incrementing probabilities



- Decrementing probabilities



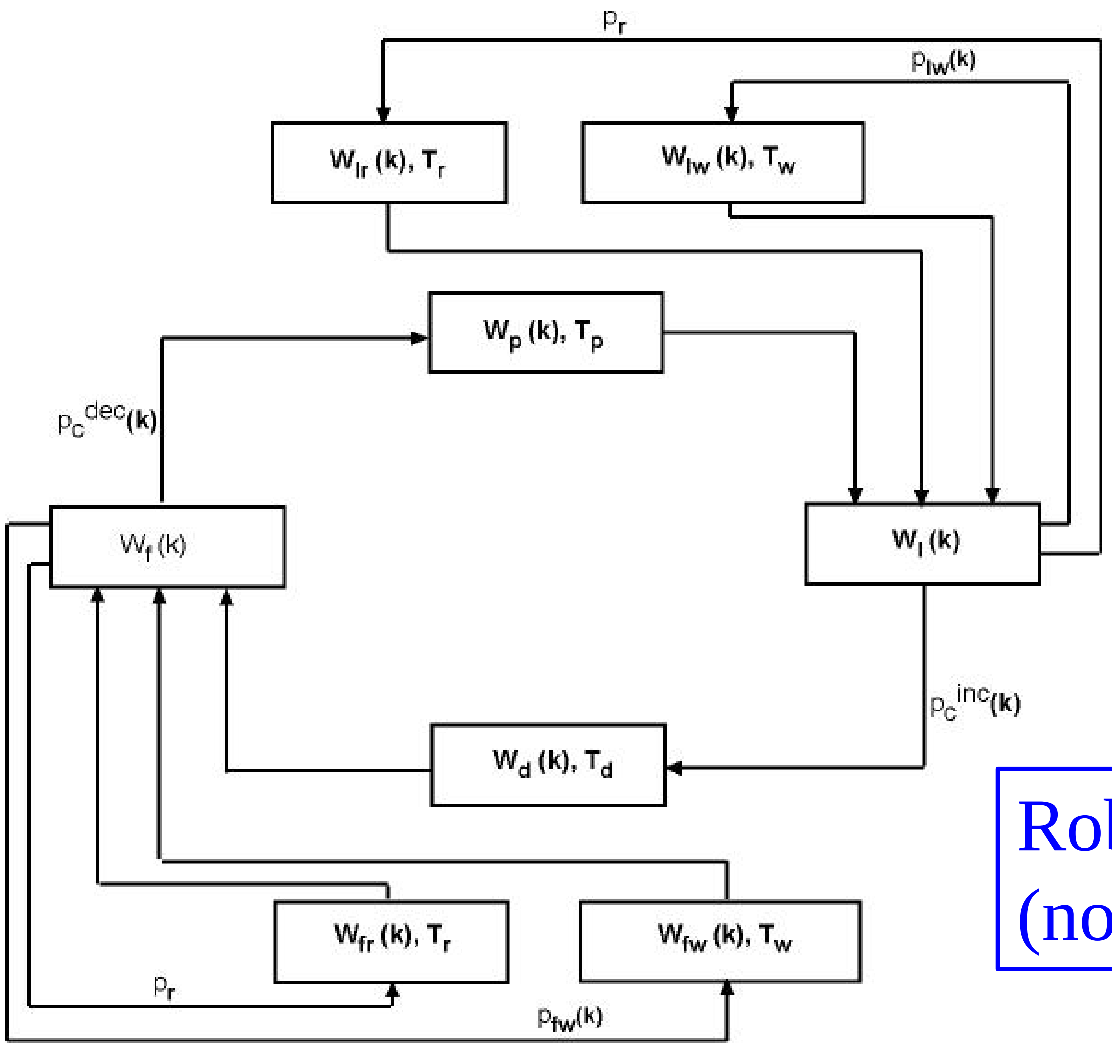
Resulting Probabilities



Perimeters are relevant for computing the cluster modifying probabilities: robot turns on the spot for object distinction before approaching the cluster!

Micro-Macroscopic Models

From Agassounon et al, 2004 (restructured representation):



Robots always active
(no worker allocation)

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Models: Explanations and Predictions

Single cluster? **All models predicted yes and in roughly how much time!**

Number of clusters (inter-distance between seeds < 1seed) monotonically decreases if:

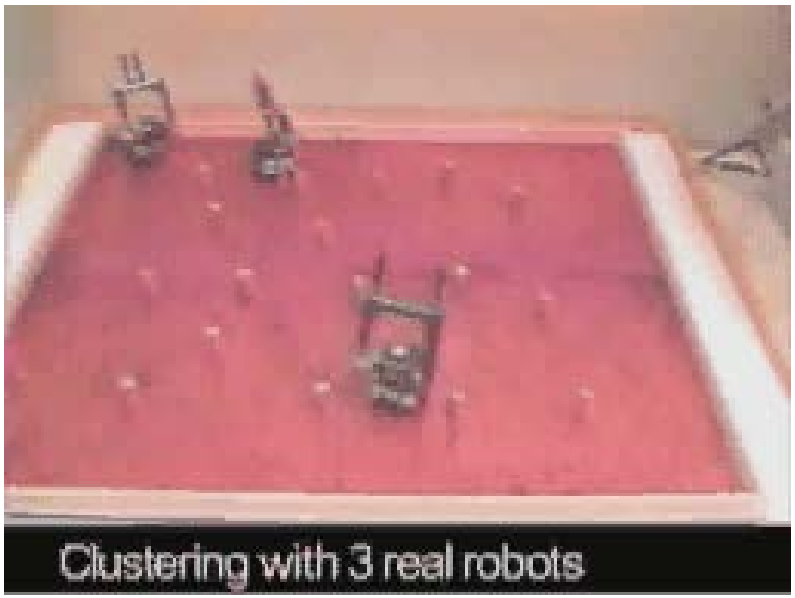
- Probability to create a NEW cluster of 1 seed in the middle of the arena is equal to zero
- No hard partitioning of the arena (robot homogeneously mix clusters)
- Cluster are not broken in two parts by removing one seed in the middle

Long Distributed Building Experiments

Submicroscopic Model
(Webots)

- 10% white noise on all sensor and actuators
- Perfectly homogeneous team
- Kinematic mode

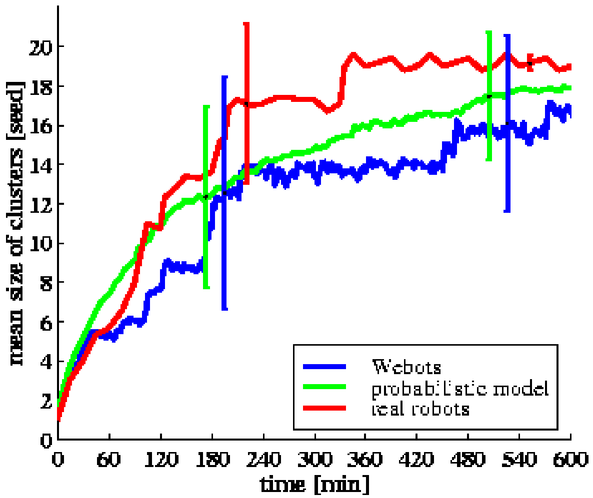
Real robots
(Khepera)

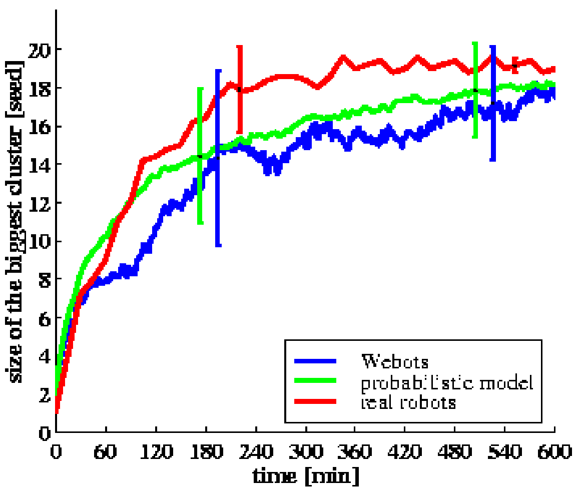


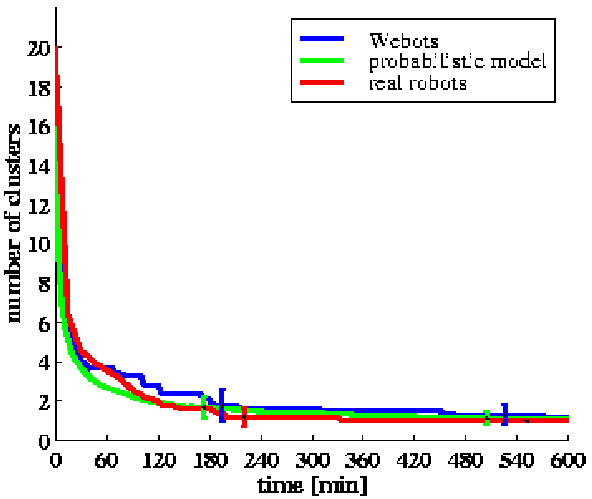
- Electrical floor: continuous power supply in any position and orientation
- Heterogeneities among teammates and components
- Inaccuracies in acting and sensing
- Dynamics (e.g., friction) plays a role

Results (till single cluster)

- 3 robots
- real robots (5 runs), submicroscopic (10 runs), microscopic model (100 runs)
- [Martinoli, Ijspeert, Mondada, 1999]

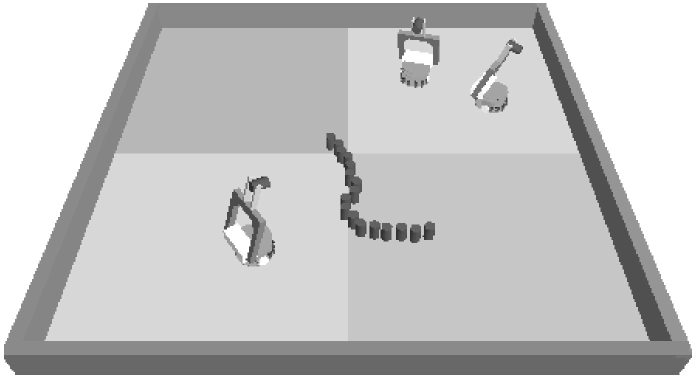






- **Mean size of clusters**
- **Size of the biggest cluster**
- **Number of clusters**

Example of arising 2D Structures



Noise in S&A and poor navigation capabilities do not allow for precise, controllable structure building

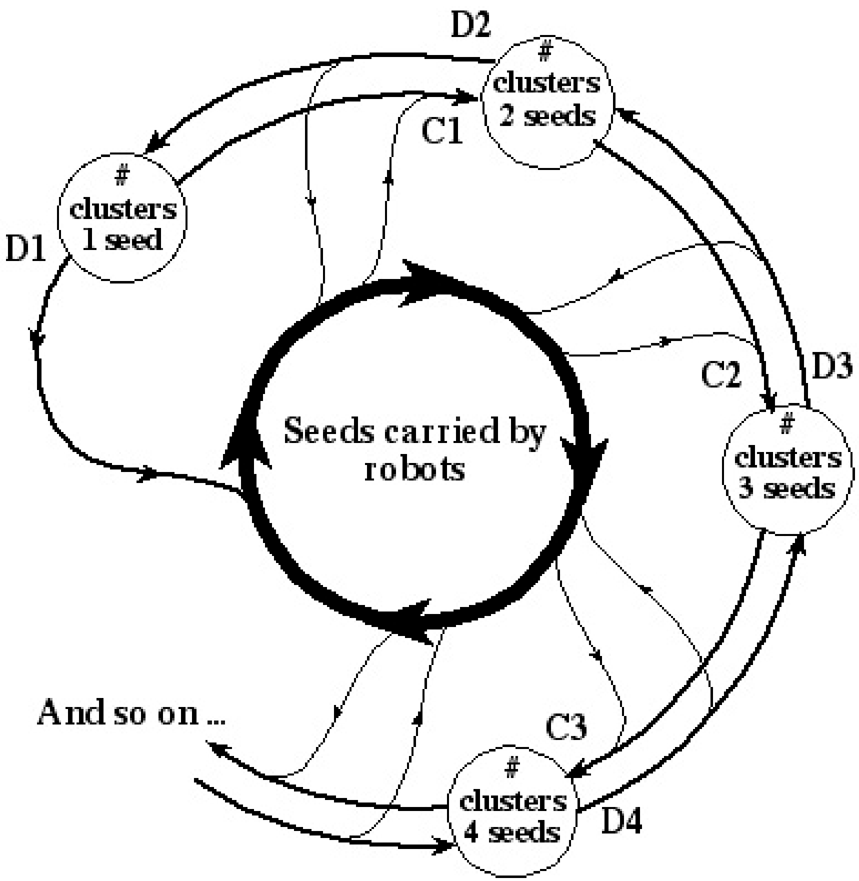
Submicroscopic

Real robots

Macroscopic Model: Distributed Building Dynamics

- $d_i(k)$ = decr_geom_probability$_i$*p_find$_i(k)$
- $c_i(k)$ = incr_geom_probability$_i$*p_find$_i(k)$

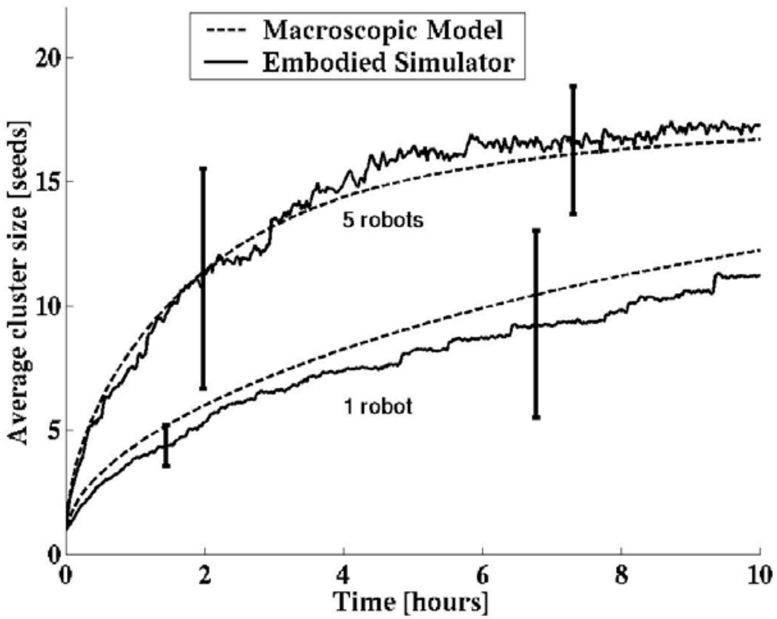
- p_find$_i(k)$ = finding probability of all the cluster of size i
- If n = number of seeds -> macroscopic model of environment with n nonlinearly coupled ODE (n for each possible cluster size) + robot states



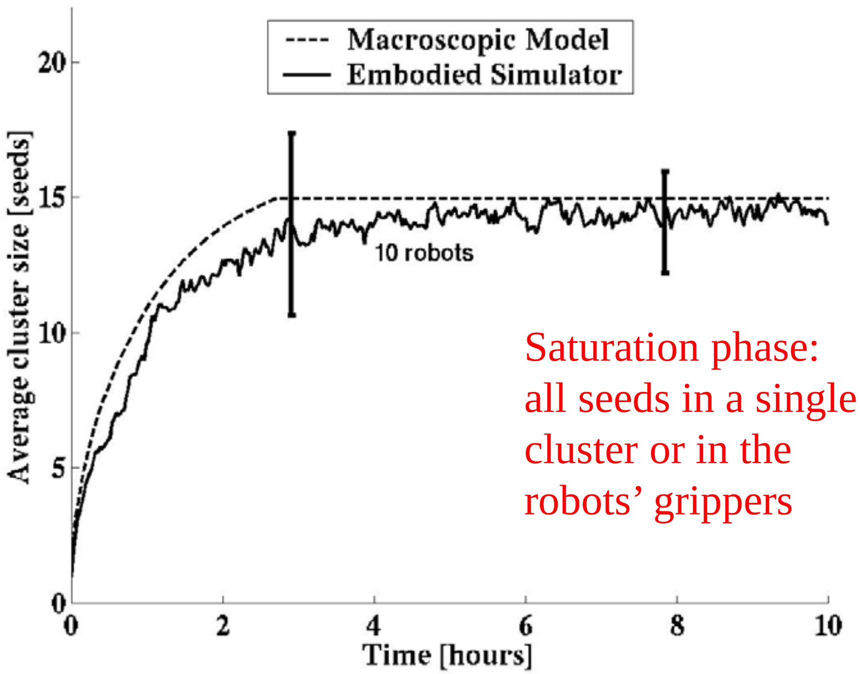
[Agassounon et al, *Autonomous Robots*, 2004]

Some Results from Agassounon et al., 2004 (1, 5, 10 robots always active)

Metric: average cluster size (20 seeds)



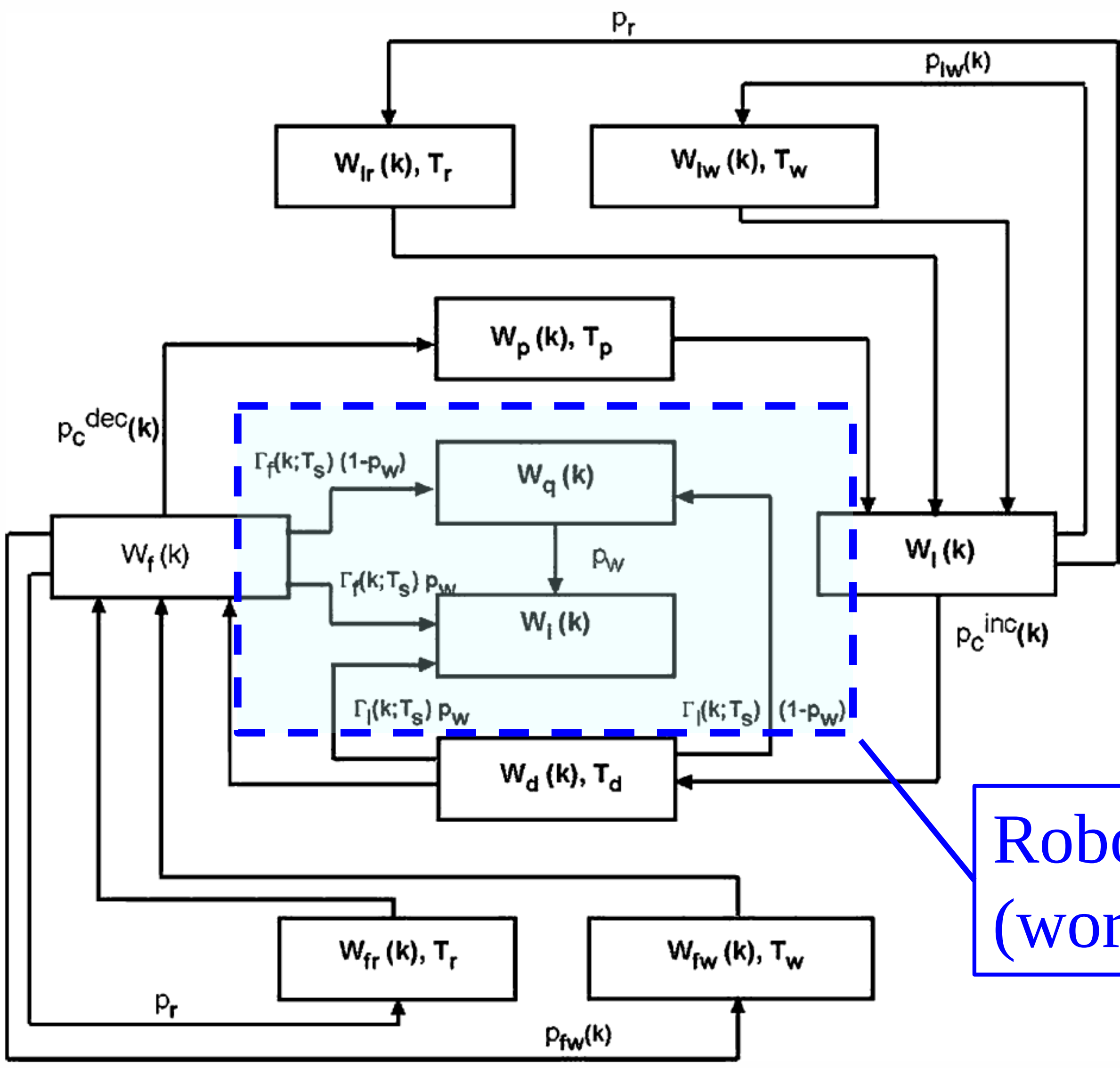
1 and 5 robots



Saturation phase: all seeds in a single cluster or in the robots' grippers

10 robots

Micro-Macroscopic Models

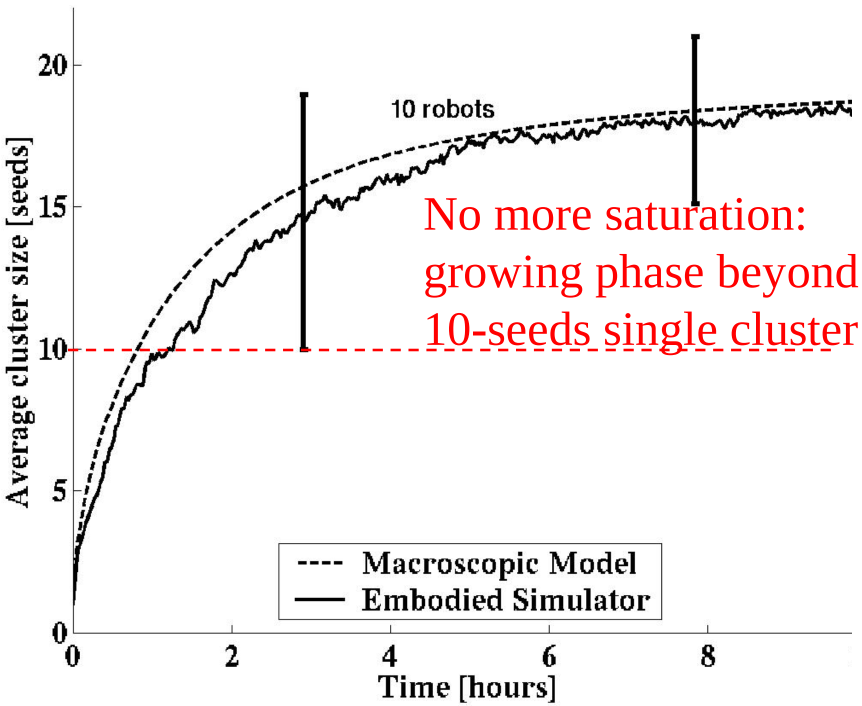


Note: see also Week 6

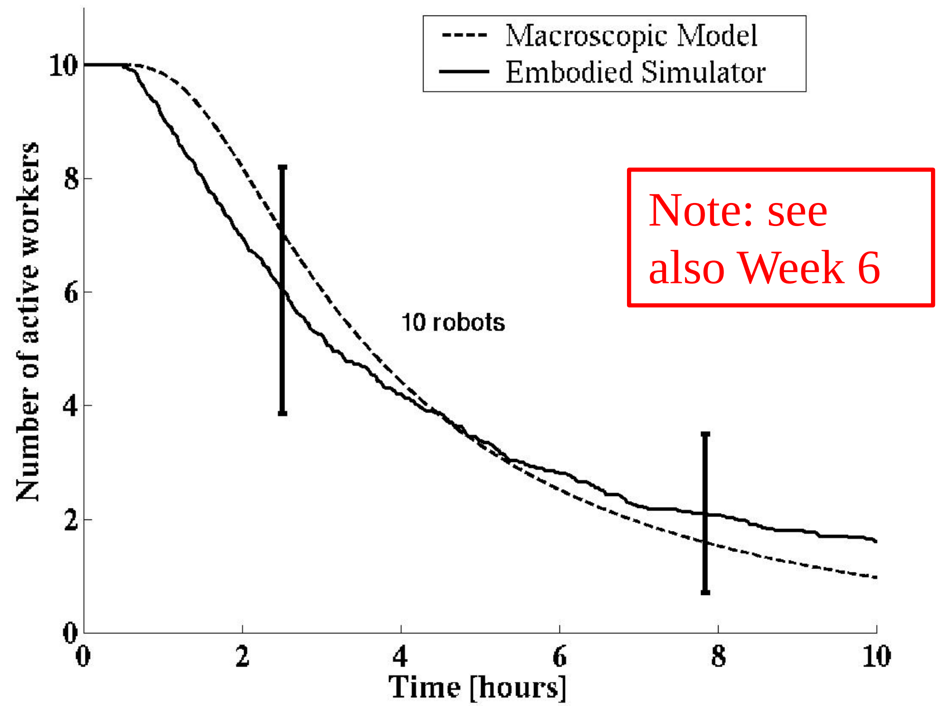
Robots can go resting (worker allocation)

Some Results from Agassounon et al., 2004 (10 robots with activity regulation)

20 seeds, threshold for abandoning the arena= 25 min, 10 robots



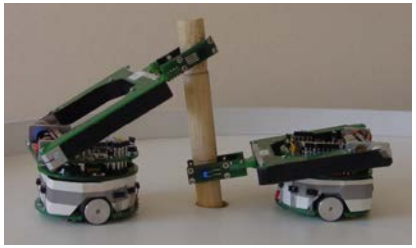
Average cluster size



Number of active robots

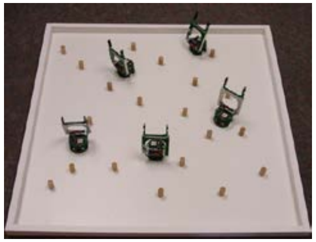
Journal Publications using the Same Modeling Framework

Stick Pulling



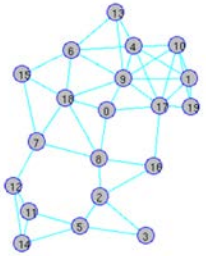
- [Martinoli, Easton, Agassounon, *Int. J. of Robotics Res.*, 2004]
- [Lerman, Galstyan, Martinoli, Ijspeert, *Artificial Life*, 2001]
- [Ijspeert, Martinoli, Billard, Gambardella, *Auton. Robots*, 2001]

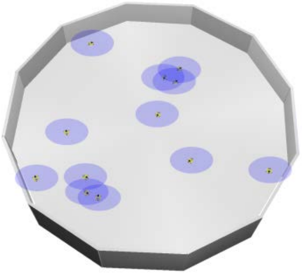
Object Aggregation



- [Agassounon, Martinoli, Easton, *Autonomous Robots*, 2004]
- [Martinoli, Ijspeert, Mondada, *Robotics and Auton. Systems* 1999]

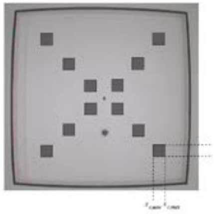
Robot Aggregation and Swarming – more on Week 14





- [Correll and Martinoli, *Int. J. of Robotics Res.*, 2011]
- [Winfield, Liu, Nembrini, Martinoli, *Swarm Intelligence J.*, 2008]

Coverage – use spatial models



- [Prorok, Correll, and Martinoli, *Int. J. of Robotics Res.*, 2011]

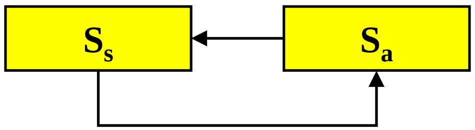
ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Combined Modeling and Machine-Learning Methods

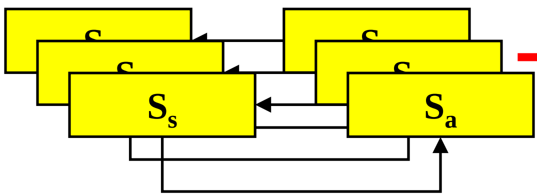
Rationale for Combined Methods (1)

- Any level of modeling (submicro, micro, or macro) allow us to consider certain parameters and leave others; models, as expression of reality abstraction, can be considered as more or less coarse "filters" of the reality
- Combined modeling/machine-learning techniques can be used at any of the abstraction levels; machine-learning techniques will explore the design parameters explicitly represented at a given level of abstraction
- Depending on the features of the hyperspace to be searched (size, continuity, noise, etc.), appropriate machine-learning techniques should be used (e.g., hill-climbing vs. population-based
- One particular optimization problem is system identification: the performance to optimize is the matching with the reality (or with a lower abstraction level). See model calibration in [Correll & Martinoli, DARS 2006].

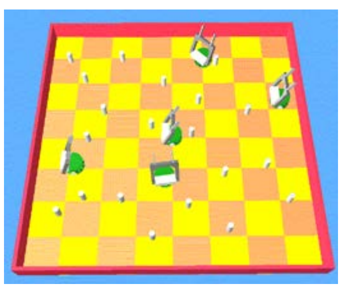
Rationale for Combined Methods (2)



Macroscopic + ML? Most of the time not needed since very fast + continuous; homogeneous systems mainly; standard numerical optimization techniques/systematic search can be used



Microscopic + ML (see this lecture's examples); for instance, diversity and specialization can be studied



Submicroscopic + ML (see Week 10 and 11 examples using PSO); for instance low-level design parameters can be learned

Target system + ML = adaptation with HW in the loop
(on-board or off-board)

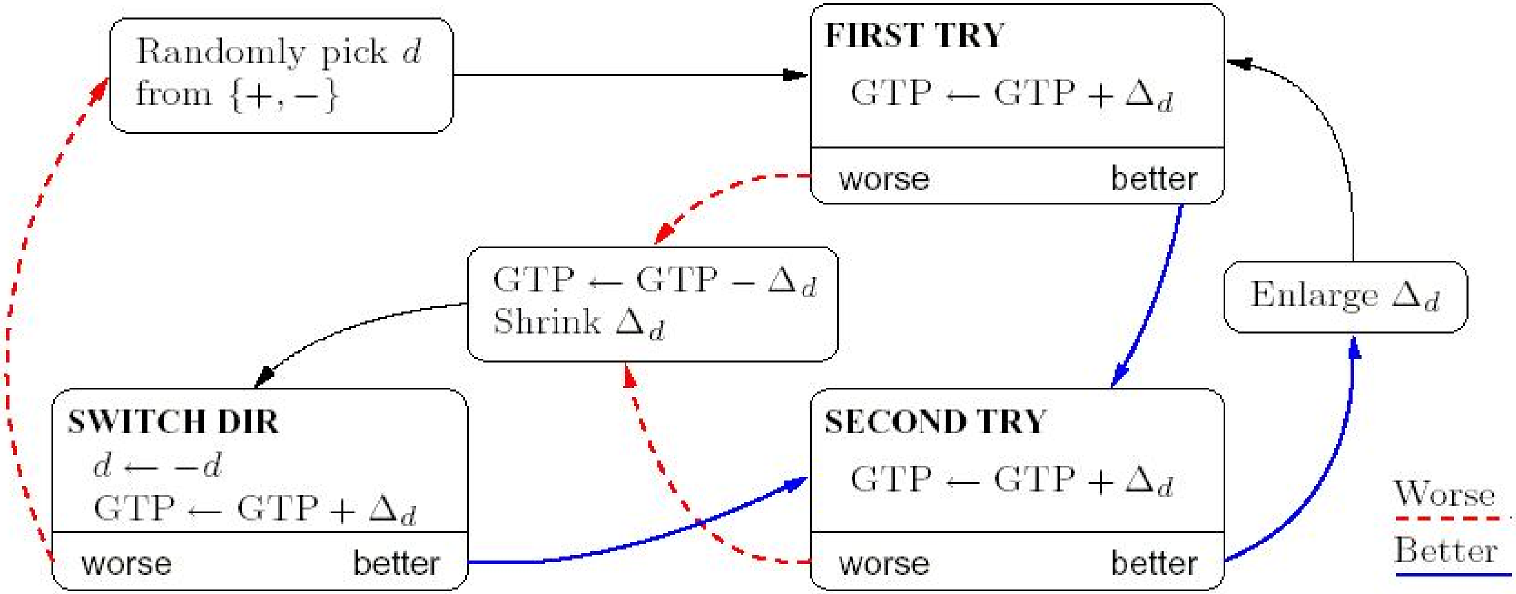
Cost of optimization/design

Abstraction

In-Line Adaptive Learning

In-Line Adaptive Learning
(Li, Martinoli, Abu-Mostafa, 2001)

- GTP: Gripping Time Parameter
- Δd: learning step
- d: direction
- Underlying low-pass filter for measuring the performance



Algorithm Parameters

Algorithmic parameters:

	Value	Description
T_m	2400	averaging period for reinforcement signal (sec)
E	1.9	GTP offset enlarge factor
F	0.3	GTP factor enlarge ratio
U	2	GTP offset shrink divider
V	0.5	GTP factor shrink ratio

Low-pass filter

Adapting rules for the learning step

From Li et al., *Adaptive Behavior*, 2004

In-Line Adaptive Learning

Differences with gradient descent methods:

- Fixed rules for calculating step increase/decrease → limited descent speed → no gradient computation → more conservative but more stable
- Randomness for getting out from local minima (no momentum)
- Underlying low-pass filter is part of the algorithm

Differences with Reinforcement Learning:

- No learning history considered (only previous step)

Differences with basic In-Line Learning:

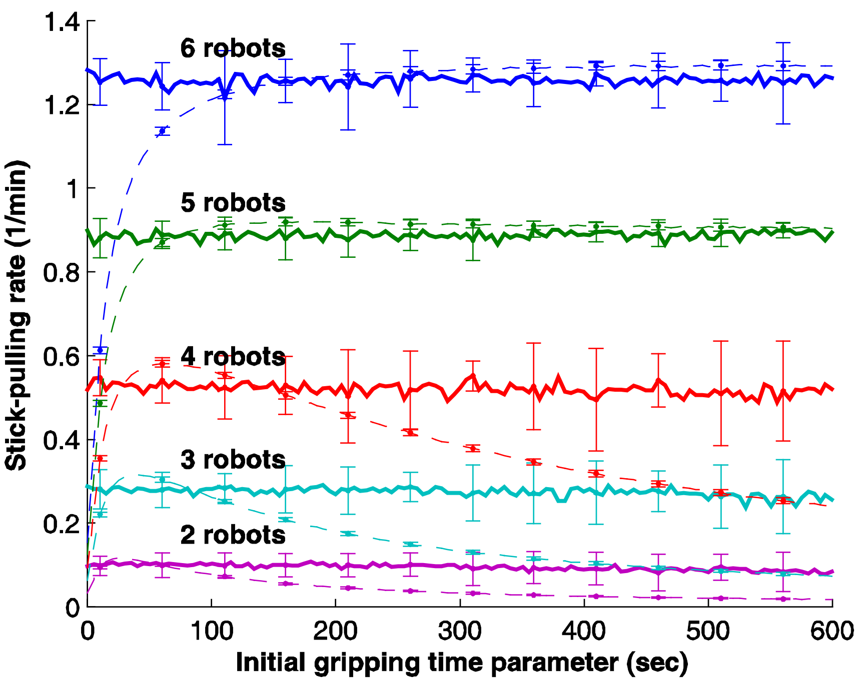
- Step adaptive → faster and more stability at convergence

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

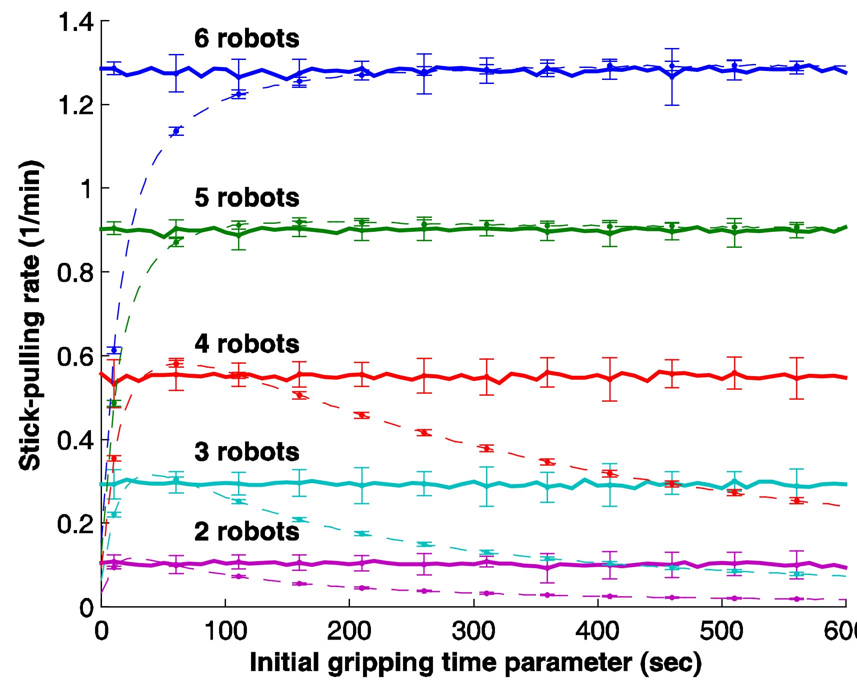
Co-Learning in a Collaborative Framework

Sample Results – Homogeneous Learning

Short averaging window
(filter cut-off *f* high)



Long averaging window
(filter cut-off *f* low)



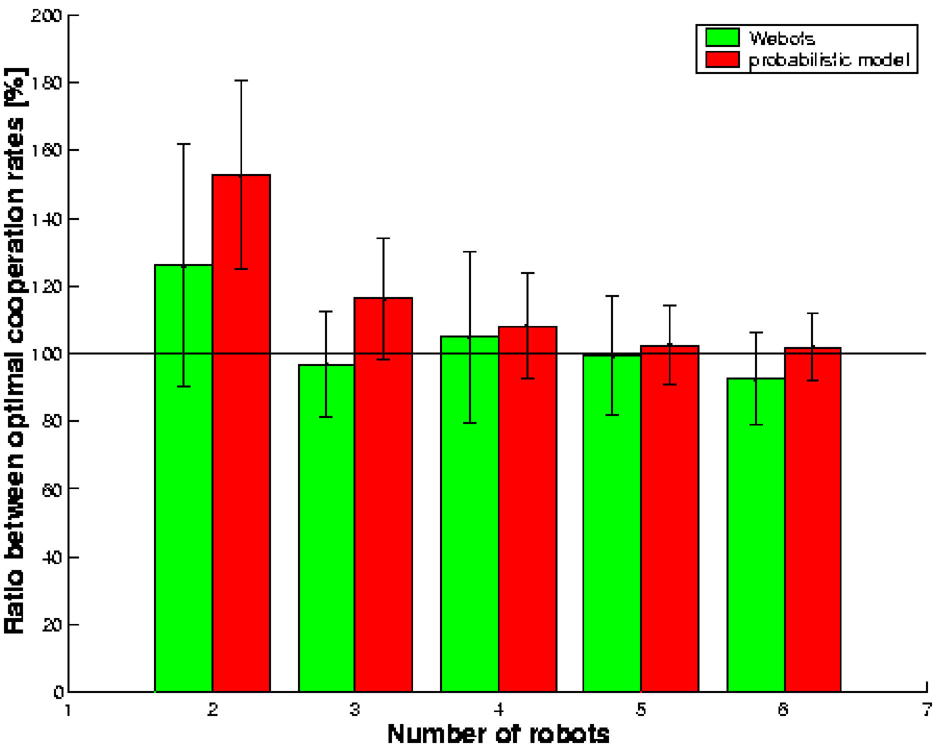
- - - - - Systematic (mean only)

——— Learned (mean + std dev)

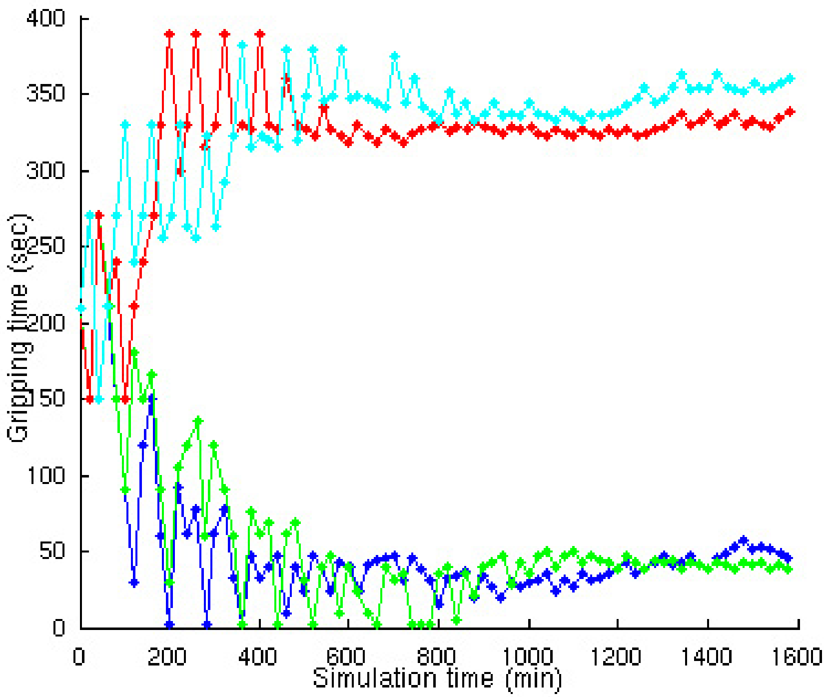
Note: 1 parameter for the whole group!

Heterogeneous Learning

Key question: does team diversity enhance performance? I.e., can individual members become specialized?



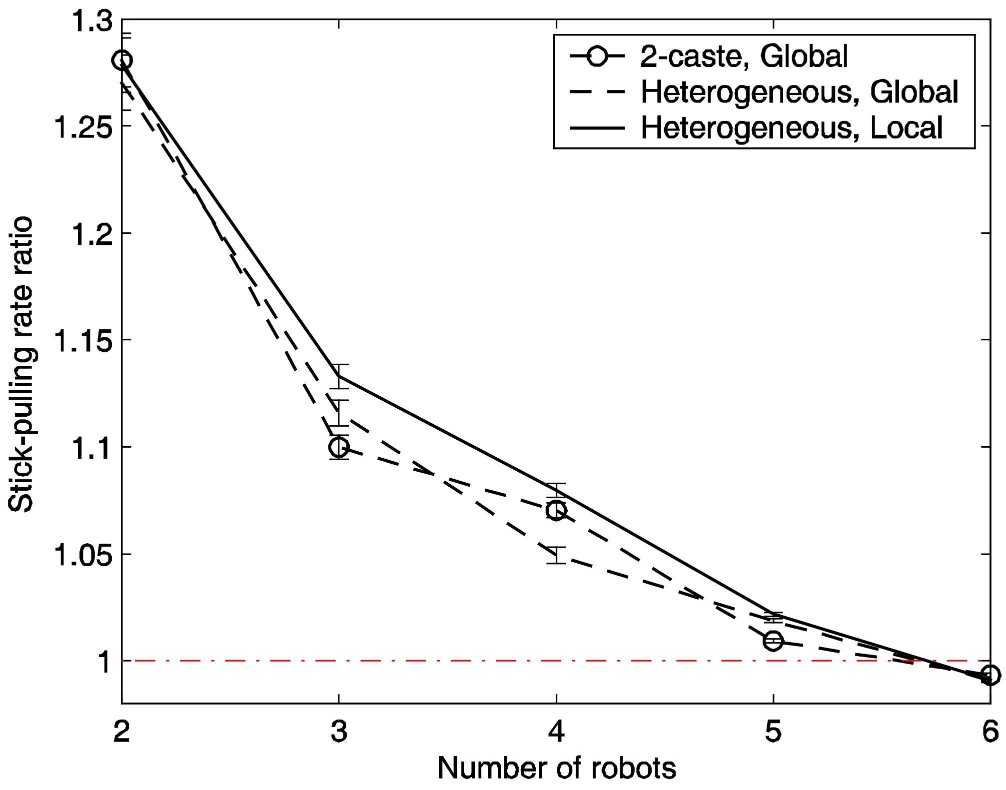
Performance **ratio** between 2 caste and homogeneous system (submicro/micro models, systematic search)



4 robots, one per color, micro + learning

Heterogeneous vs. Homogenous Learning

[Li et al., *Adaptive Behavior*, 2004]



Notes:

- large T_m (long averaging window)
- only private strategies
- global = group
  local = individual

Performance **ratio** between heterogeneous (full and 2-castes) and homogeneous groups AFTER learning

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Measuring Diversity and Specialization

Diversity Metrics
(Balch 1998)

Entropy-based diversity measure introduced in AB-04 could be used for analyzing threshold distributions

Simple entropy: $H(\Re) = -\sum_{i=1}^{m} p_i \log p_i.$ Social entropy: $D(\Re) = \int_0^{\infty} H(\Re, h)\mathrm{d}h.$

p_i = portion of the agents in cluster i; m cluster in total; h = taxonomic level parameter

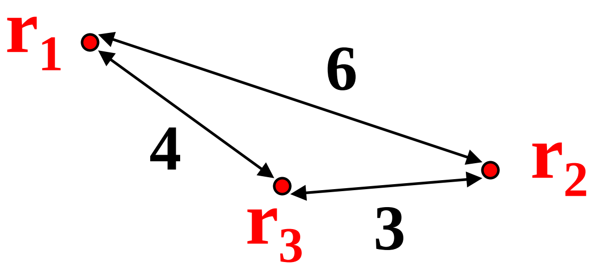
---

Input: a swarm system $\Re = \{r_1, r_2, \ldots, r_n\}$ of size n; a difference measure d.
For different level h, the C_u clustering algorithm does:

1. Initialize n clusters with cluster $c_i = \{r_i\}$;
2. For each c_i: for each r_j: If $d(r_j, r_k) \leq h$ for all r_k in c_i, add r_j to cluster c_i;
3. Discard redundant clusters;
4. Calculate p_i and the entropy $H(\Re, h)$. Note that when r_j belongs to s clusters including c_i, its contribution to p_i is $1/sn$.

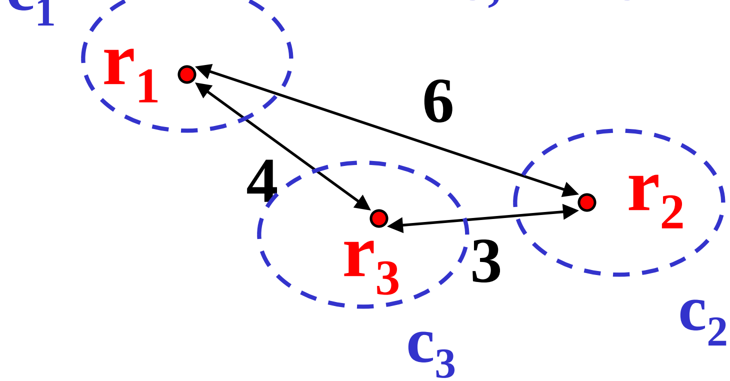
Return $\int_0^{\infty} H(\Re, h)\, \mathrm{d}h$ as the hierarchic social entropy.

Example – Simple Entropy



- R = {r_1, r_2, r_3}
- n = 3 (three swarm points)
- bi-dimensional space
- define a distance: Euclidian distance
- h = taxonomic level parameter
- m = number of clusters

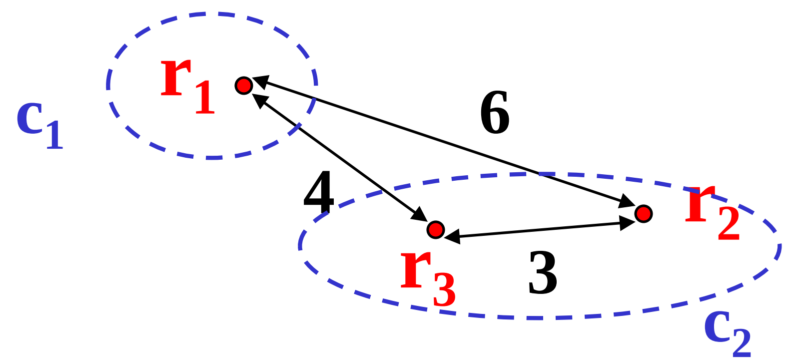
h < 3, m = 3



$$H(R) = -\sum_{i=1}^{3} p_i \log p_i = H(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}) =$$

$$-3\frac{1}{3}\log\frac{1}{3} = 0.477$$

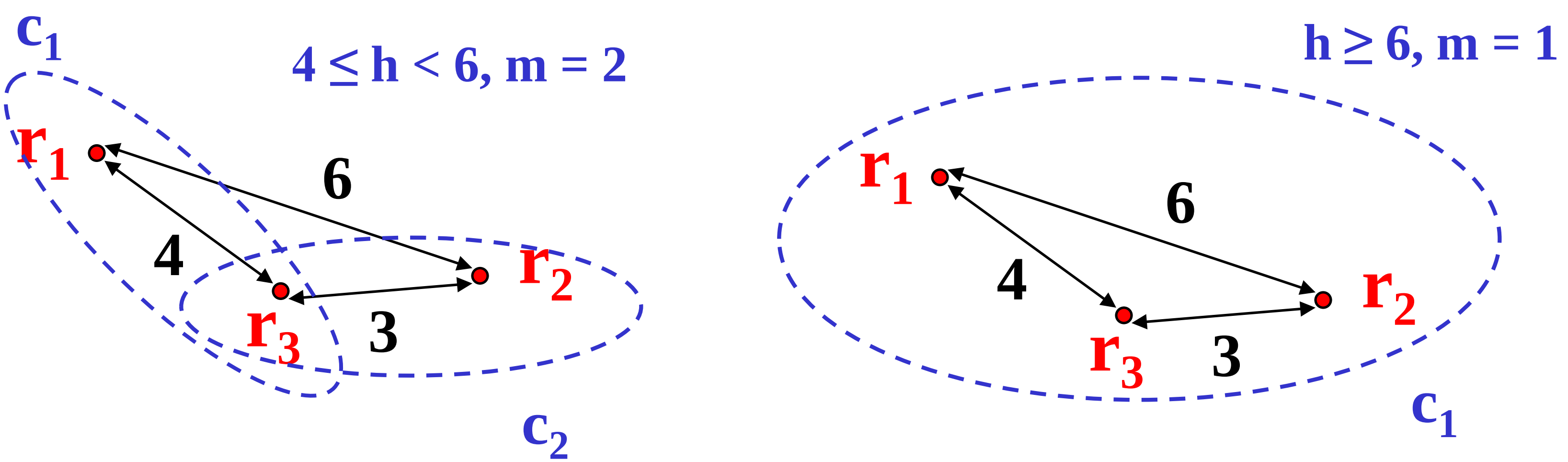
3 ≤ h < 4, m = 2



$$H(R) = -\sum_{i=1}^{2} p_i \log p_i = H(\frac{1}{3}, \frac{2}{3}) =$$

$$-\frac{1}{3}\log\frac{1}{3} - \frac{2}{3}\log\frac{2}{3} = 0.159 + 0.117 = 0.276$$

Example – Simple Entropy



$$H(R) = -\sum_{i=1}^{2} p_i \log p_i = H\left(\frac{1}{3} + \frac{1}{3}\frac{1}{2}, \frac{1}{3} + \frac{1}{3}\frac{1}{2}\right) = H\left(\frac{1}{2}, \frac{1}{2}\right)$$

$$-\frac{1}{2}\log\frac{1}{2} - \frac{1}{2}\log\frac{1}{2} = 0.301$$

$$H(R) = -\sum_{i=1}^{1} p_i \log p_i = H\left(\frac{3}{3}\right) = -\log 1 = 0$$

Check $\sum_{i=1}^{m} p_i = 1$ with overlapping clusters!

Example – Social Entropy

$$D(R) = \int_0^\infty H(R,h)dh = 3\times H(\frac{1}{3},\frac{1}{3},\frac{1}{3}) + 1\times H(\frac{1}{3},\frac{2}{3}) + 2\times H(\frac{1}{2},\frac{1}{2}) + 0 = 2.309$$

Note: In contrast to simple entropy ≥ 1

Contrast with R = {r1, r2, r3} and $r_1 = r_2 = r_3$ (homogeneous swarm), for any h $\geq 0 \rightarrow$ single cluster $\rightarrow$ D(R) = 0!

Differences with Plain Euclidian Diversity Measure

$$d(a,b) = \sqrt{\sum_i (a_i - b_i)^2}$$

Components in all dimensions

$$D_{eu} = \frac{1}{N(N-1)} \sum_a \left[\sum_{b \neq a} d(a,b) \right]$$

All points from any other point

- Underlying distance measure in the solution space might be the same (e.g. Euclidian distance)
- Social entropy is looking for possible clustering of the vectors (looking for possible castes) while Euclidian diversity is just looking how spread out/diverse in general are the vectors

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Specialization Metric

Specialization metric introduced in AB-04:

$$S = \text{corrcoef}(D; R) \times D.$$

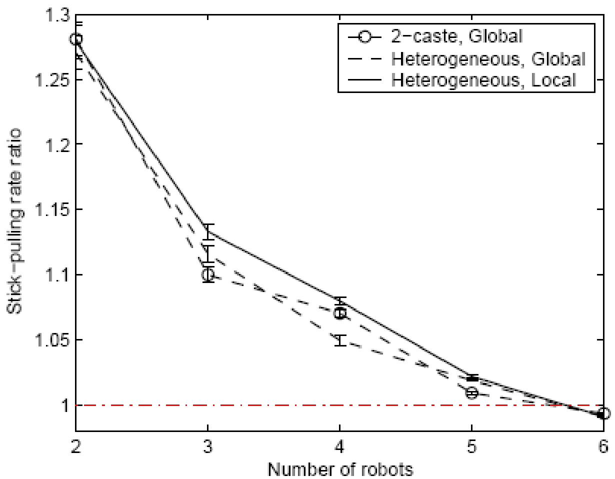
S = specialization; D = diversity (e.g., social entropy); R = swarm performance

Notes

- Idea: “weighting diversity with performance”
- This is useful when the number of tasks to be solved is not well-defined or it is difficult to assess the task granularity a priori. In such cases the mapping between task granularity and caste granularity might not trivial (see the limited performance of a caste-based solution in the stick-pulling experiment)
- Could be used for analyzing specialization arising from a variable-threshold division of labor algorithm (see lecture Week 6)

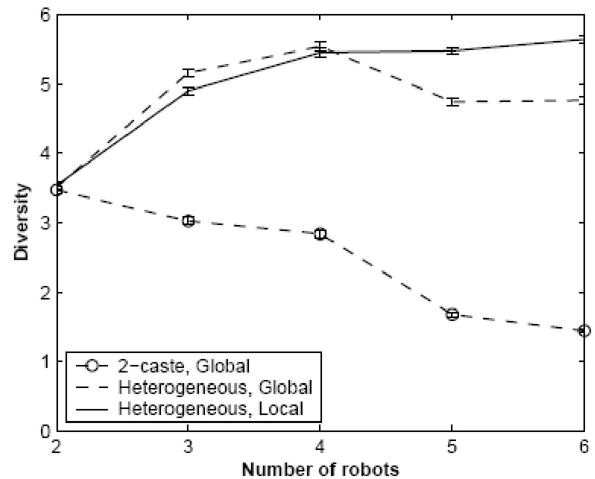
Sample Results in the Standard Sticks

- 2 serial grips needed to get the sticks out
- 4 sticks, 2-6 robots, 80 cm arena



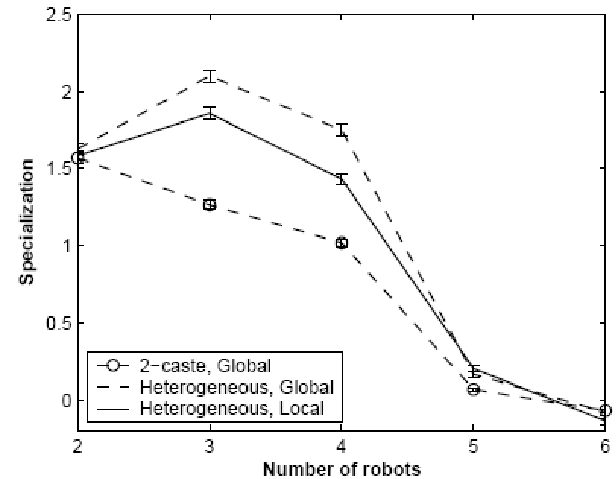
Relative Performance

- Specialists more important for small teams
- Local p > global p
- Enforced caste: pay the price for odd team sizes



Diversity

- Measured using social entropy
- Flat curves, difficult to tell whether diversity bring performance



Specialization

- Specialization higher with global when needed, drop more quickly when not needed
- Enforcing caste: “low-pass filter” effect

Conclusion

Take Home Messages

- The multi-level modeling methodology is a framework that has been successfully used in multiple case studies
- Models' parameter calibration is difficult and still an open challenge
- An additional case study has illustrated how to capture time-varying parameters depending on the environmental modifications introduced by the robots and how to choose an appropriate state granularity
- Different modeling levels can be combined with machine-learning for design and optimization purposes
- Microscopic models allows for efficiently studying diversity and specialization issues
- Specialization is the part of diversity that improves performance
- The diversity and specialization level of a heterogeneous swarm can be quantitatively measured

Additional Literature – Week 9

Papers

- Prorok A., Correll N., and Martinoli A., "Multi-level Spatial Modeling for Stochastic Distributed Robotic Systems". Special issue on Stochasticity in Robotics and Biological Systems, Asada H. H. and Kumar V. editors, *Int. Journal of Robotics Research*, 30(5): 574-589, 2011.
- Di Mario E., Mermoud G., Mastrangeli M., and Martinoli A. "A Trajectory-based Calibration Method for Stochastic Motion Models". *Proc. of the 2011 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, September 2011, San Francisco, U.S.A., pp. 4341-4347.
- Roduit P., Martinoli A., and Jacot J., "A Quantitative Method for Comparing Trajectories of Mobile Robots Using Point Distribution Models". *Proc. of the 2007 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, October-November 2007, San Diego, USA, pp. 2441-2448.
- Correll N. and Martinoli A., "System Identification of Self-Organizing Robotic Swarms". *Proc. of the Eight Int. Symp. on Distributed Autonomous Robotic Systems*, July 2006, Minneapolis/St. Paul, MN, U.S.A. Distributed Autonomous Robotic Systems 7 (2006), pp. 31–40. Best Paper Award.
- Martinoli A., Ijspeert A. J., and Mondada F., "Understanding Collective Aggregation Mechanisms: From Probabilistic Modelling to Experiments with Real Robots". Special issue on Distributed Autonomous Robotic Systems, Dillmann R., Lüth T., Dario P., and Wörn H., editors, *Robotics and Autonomous Systems*, 29(1): 51-63, 1999.
- Ijspeert A. J., Martinoli A., Billard A., and Gambardella L.M., "Collaboration through the Exploitation of Local Interactions in Autonomous Collective Robotics: The Stick Pulling Experiment". *Autonomous Robots*, **11**(2):149–171, 2001.
- Murciano, A., Millán, J. del R., & Zamora, J. (1997). Specialization in multi-agent systems through learning. *Biological Cybernetics*, *76*, 375–382.
- Wolpert, D. H., & Tumer, K. (2001). Optimal payoff functions for members of collectives. *Advances in Complex Systems*, *4*, 265–279.
- Winfield A. F. T., Liu W., Nembrini J., and Martinoli A., "Modelling a Wireless Connected Swarm of Mobile Robots. Special issue on Swarm Robotics, Winfield A. F. T. and Sahin E., editors, *Swarm Intelligence Journal*, 2(2-4): 241-266, 2008.