Optimal foraging theory applied to swarm robotics

Alexandre Campo

`Alexandre.Campo@ulb.ac.be`

IRIDIA

Université Libre de Bruxelles

Plan

★ Introduction

★ Experimental setup

★ Simulation software

★ Analytical model

★ Autonomous behaviour & optimality

★ Conclusions

Introduction

- ★ Foraging tasks in swarm intelligence
- ★ Optimal foraging theory
- ★ Macroscopic modelling
- ★ Goal

Foraging tasks in swarm intelligence

- traditional foraging means collecting objects
- multi-foraging -> several object types
- many possible applications :
  - mine search and removal
  - search and rescue
  - collection of minerals (rover exploration)

Optimal foraging theory

- biologists hypothesized that natural selection shaped optimal foraging behaviour
- this is called optimal foraging theory (OFT)
- many predator prey models were devised for OFT
- the models and conclusions can be partly reused for optimal foraging in swarm intelligence

Macroscopic modelling

- models are useful to make faster studies, achieve mathematical analysis
- very employed in physics, biology
- many possible analytical models : rate equations models are simple and efficient
- the latter require some assumptions : homogeneity of setup and dilute events

Goal

- The goal of this work is to achieve a fully autonomous optimal behaviour in a multi foraging task.
- Tools employed are macroscopic models and simulations.

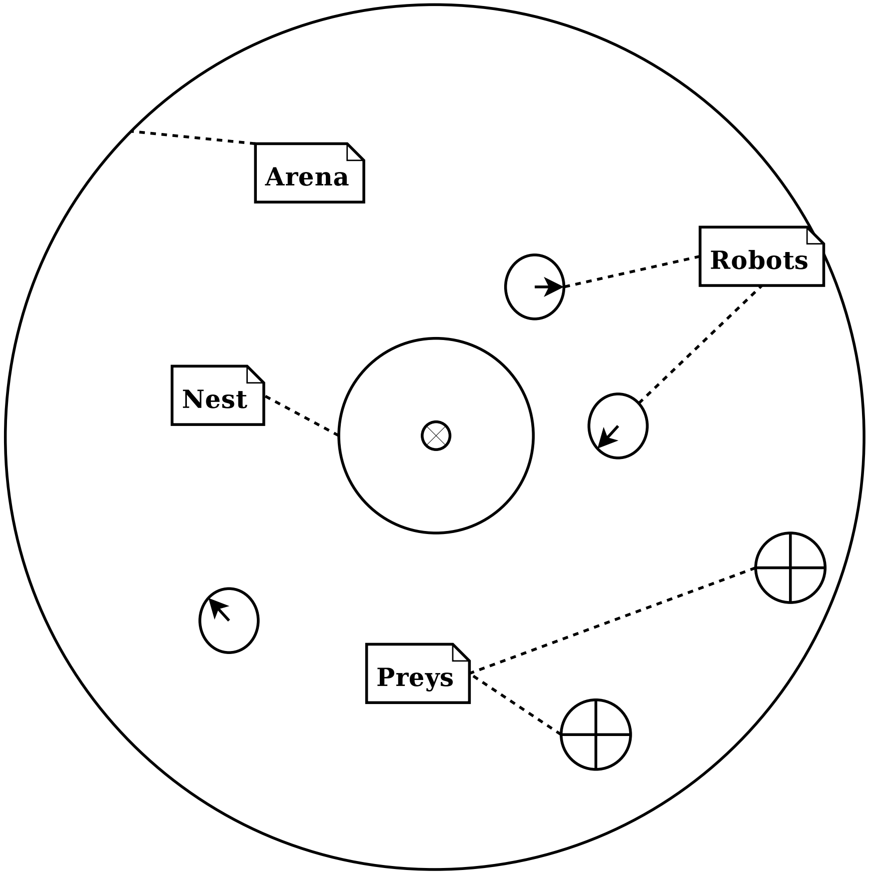
Experimental setup

- ★ Description
- ★ Nest
- ★ Prey
- ★ Robots

Experimental setup

Description

- Circular arena
- Central nest
- Robots are spread randomly in the arena
- Prey are introduced at random positions outside the nest
- There are two type of prey (different characteristics)



Experimental setup

Goal of the robots

- When retrieved to the nest, prey yield a reward
- When robots go outside the nest, they get a penalty (negative reward) for usage
- The goal of the robots is to maximize their reward by choosing :
  - $\underline{T_g}$ the proportion of time spent outside the nest
  - $\underline{C_1}$ the probability to take a prey of type 1 when encountered
  - $\underline{C_2}$ the probability to take a prey of type 2 when encountered

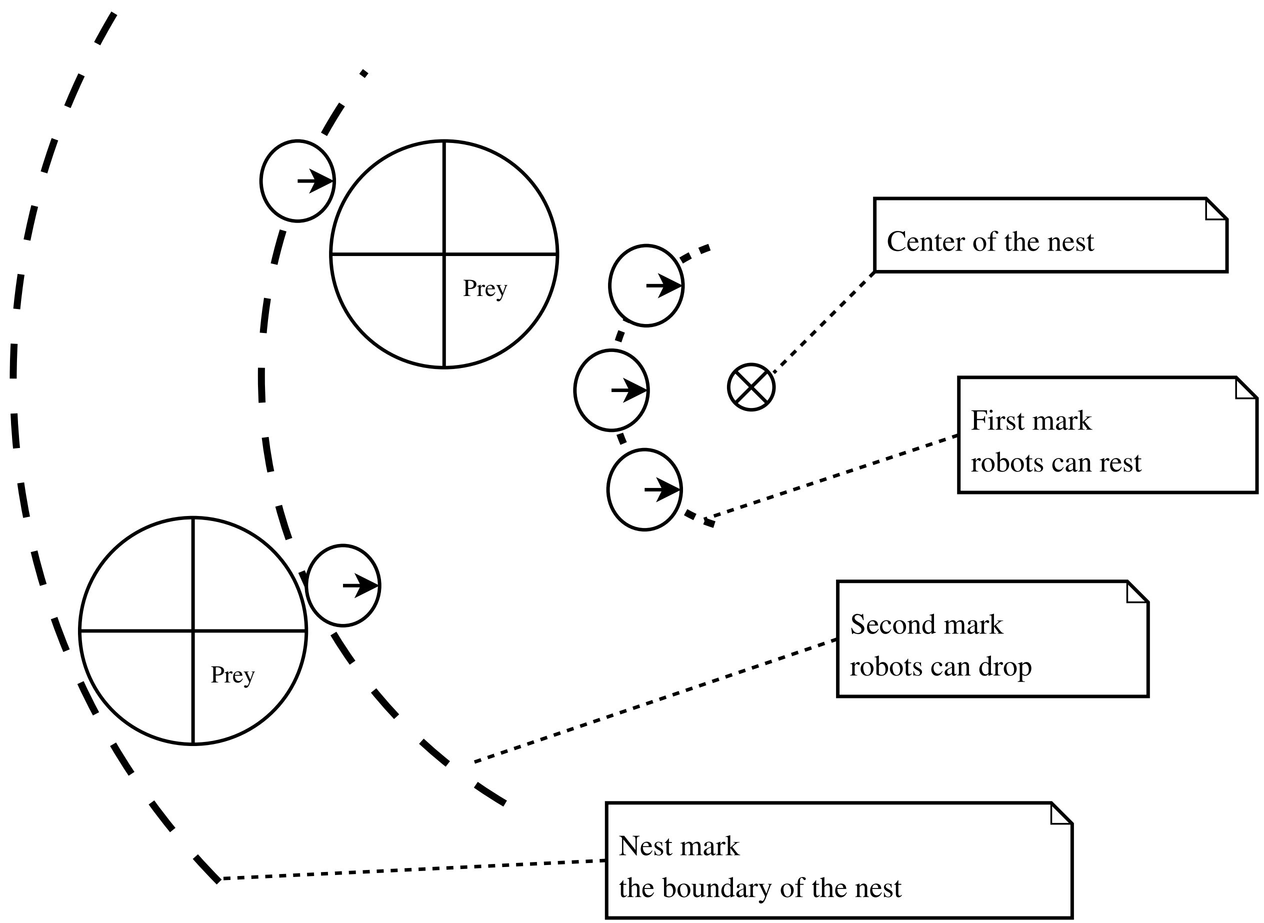
Experimental setup

Nest

- Robots can rest at nest.
- Robots must drop retrieved prey inside the nest
- The nest has 3 specific marks (to avoid overcrowding)
  - A first mark defines where robots can rest
  - A second mark defines the place where robots can release the prey
  - A last mark defines the boundary of the nest

Experimental setup

Nest



Experimental setup

Prey

- A reward for retrieval is associated to each kind of prey
- Prey can have different frictions (hence retrieval times)
- When dropped in nest, they are removed by experimentator
- Prey have a constant probability per unit time to appear at a random location at the periphery of the arena
- Prey have a lifetime.

Experimental setup

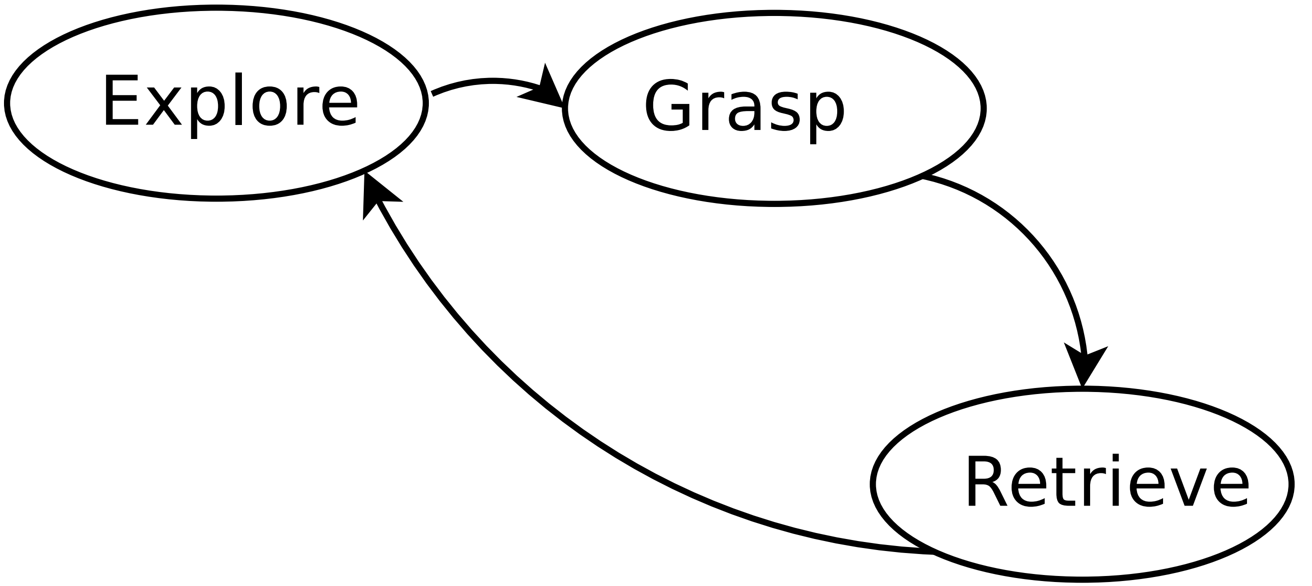
Robots

- Can perceive the nest anywhere in the arena, thanks to a lamp
- Can grasp and retrieve prey
- Can perceive robots retrieving prey (green colour)
- The controller is a finite state machine (FSA)

Experimental setup

Robots controllers

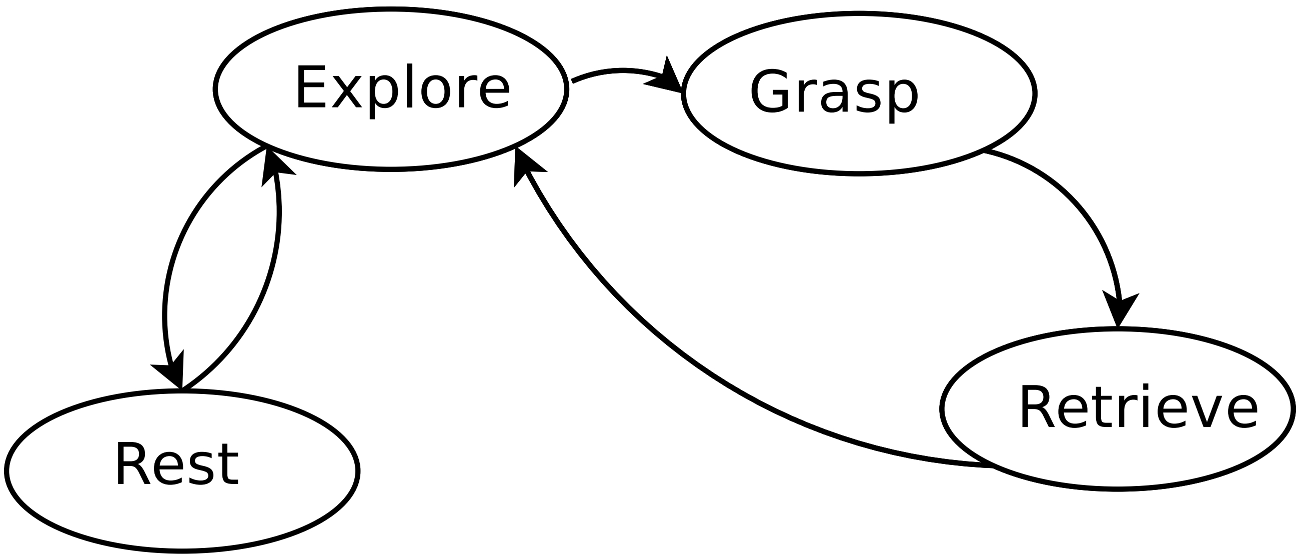
Description of the FSA used :



Experimental setup

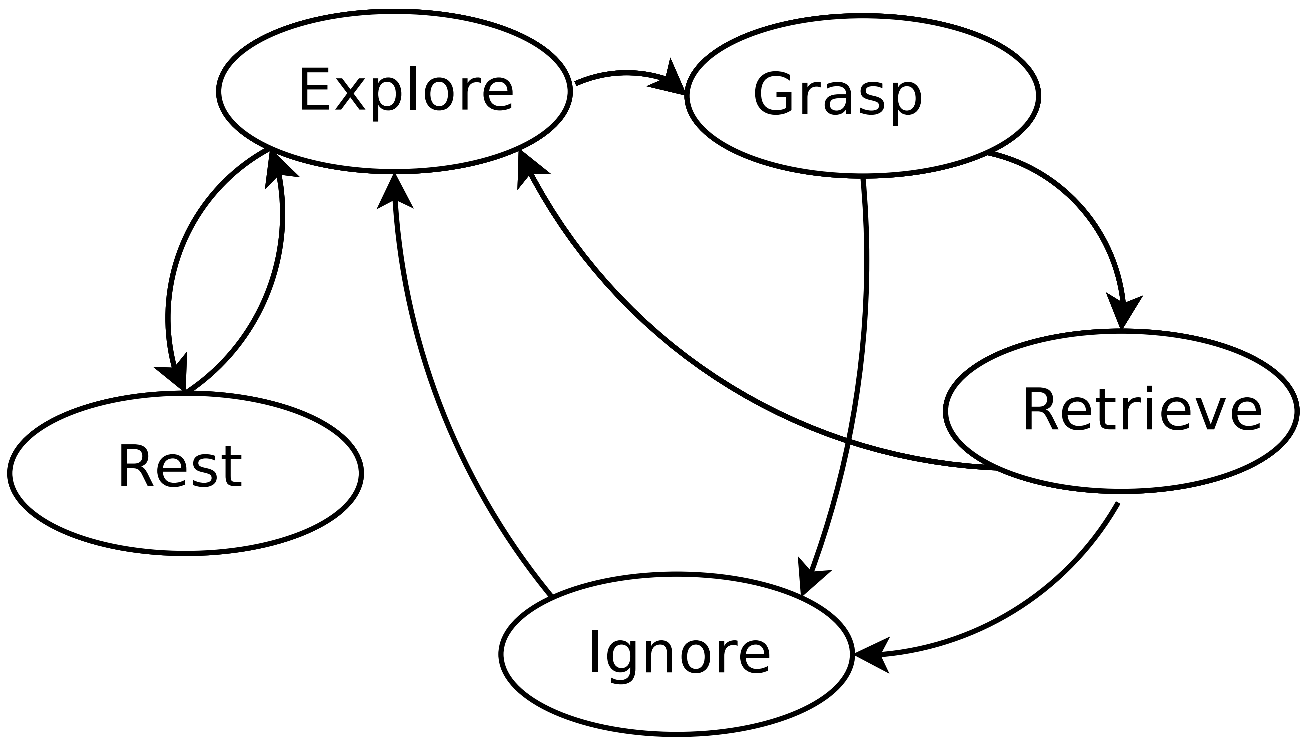
Robots controllers

Description of the FSA used :



Robots controllers

Description of the FSA used :



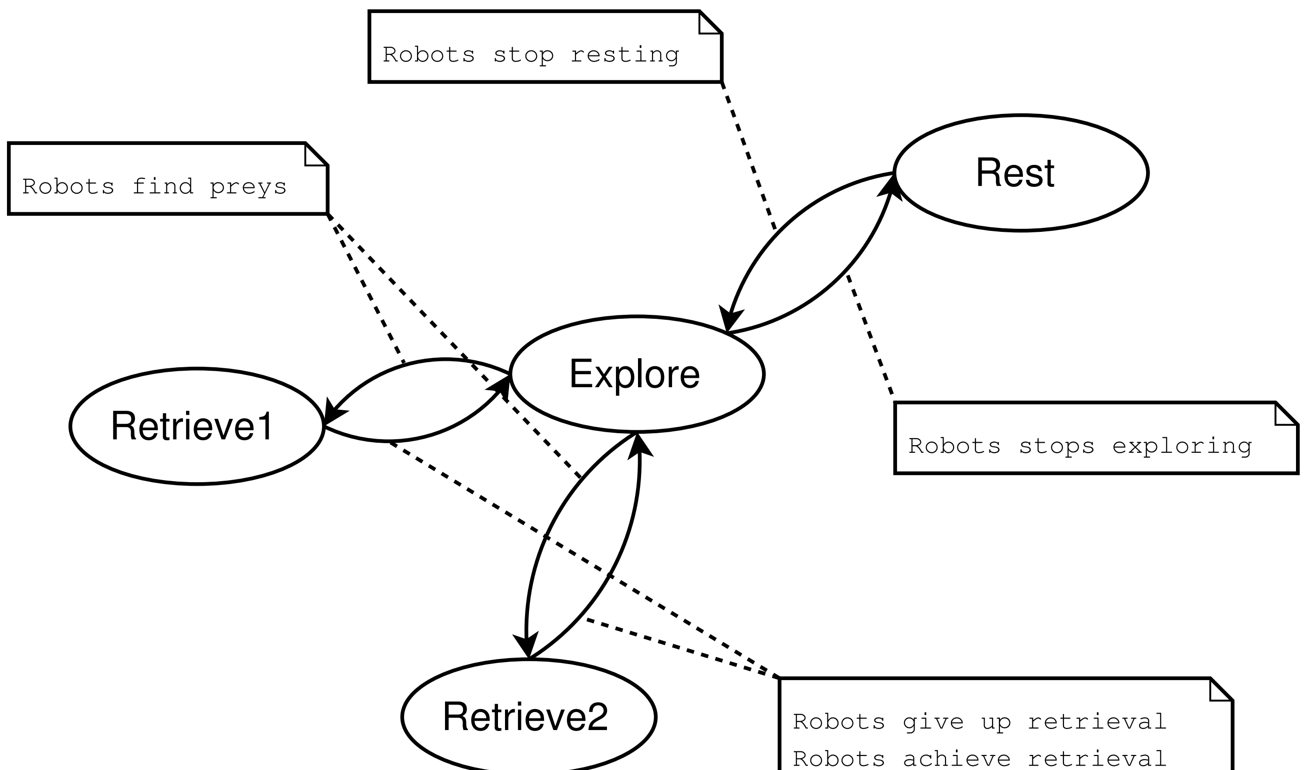
Analytical model

- ★ Flow description
- ★ Equations
- ★ Measures and calibrations
- ★ Three main cases
- ★ Validation

Analytical model

Flow description

The design of the model can be summarized by a scheme of flows between robots' states :



Analytical model

Equations

$$
\begin{aligned}
\frac{\partial F}{\partial t} &= -\beta L + \gamma I - C_1 F N_1 p_e - C_2 F N_2 p_e + \mu_1 R_1 + \mu_2 R_2 + \varepsilon R_1 + \varepsilon R_2 \\
\frac{\partial I}{\partial t} &= +\beta L - \gamma I \\
\frac{\partial R_1}{\partial t} &= C_1 F N_1 p_e - \mu_1 R_1 - \varepsilon R_1 \\
\frac{\partial R_2}{\partial t} &= C_2 F N_2 p_e - \mu_2 R_2 - \varepsilon R_2 \\
\frac{\partial N_1}{\partial t} &= \varphi_1 - C_1 F N_1 p_e - \xi_1 N_1 + \varepsilon R_1 \\
\frac{\partial N_2}{\partial t} &= \varphi_2 - C_2 F N_2 p_e - \xi_2 N_2 + \varepsilon R_2
\end{aligned}
$$

F : free robots
I : inactive robots
R_1 : robots retrieving type 1
R_2 : robots retrieving type 2
N_1 : prey of type 1
N_2 : prey of type 2

Measures and calibrations

- Most of the constants were simply set in the program (eg incoming prey rate, prey lifetime)
- The frictions of the prey were adjusted to match the retrieval probabilities μ_1 and μ_2
- p_e the probability to find a single prey for a single robot in the arena was measured in a simulation of 100000 seconds

Validation

- We generated 6000 different possible setups using parameters variations

Variable	Range of values tested	unit
R	$1, 2, 3, 5, 10, 15$	robot
$N1$	5	prey of type 1
$N2$	5	prey of type 2
$Rw1$	$-1, 1, 10, 100, 1000$	reward
$Rw2$	1	reward
Rg	-0.001	reward
β		
γ		
$giveup$	0.0111	probability
p_e	$1/166.66$	probability
μ_1	$1/90, 1/40, 1/30, 1/60$	second^{-1}
μ_2	$1/60$	second^{-1}
φ_1	$0.066, 0.033, 0.016, 0.00833, 0.0055$	prey / second
φ_2	0.0166	prey / second
ξ_1	0.002	probability
ξ_2	0.002	probability
C_1		
C_2		

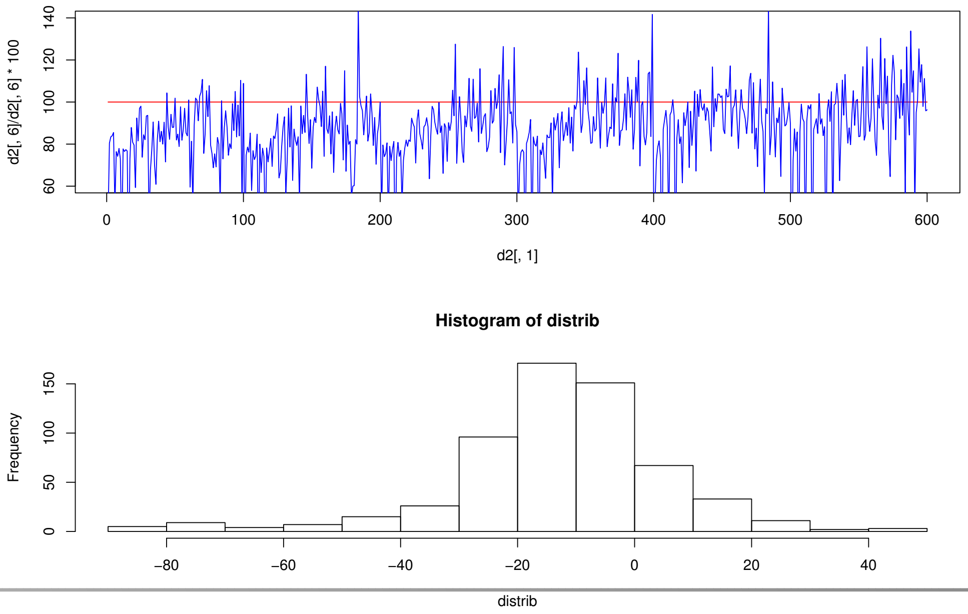
Validation

- We measure the comparison ability of the model :
- 88 % of good comparisons, noise included.

Simulation's results	$R(A) < R(B)$	$R(A) = R(B)$	$R(A) > R(B)$
Model's predictions	0%	0%	6.1%
$R(A) < R(B)$	44.45 %	0%	0%
$R(A) = R(B)$	0%	0%	0%
$R(A) > R(B)$	6.05%	0%	43.22 %

Validation

- We measure the predicted reward with respect to the outcome of simulations
- The models overestimates the reward of about 13 % (+/- 18.63 sd)



Autonomous behaviour & optimality

- ★ Algorithm
- ★ Comparison to optimal predictions

Autonomous behaviour & optimality

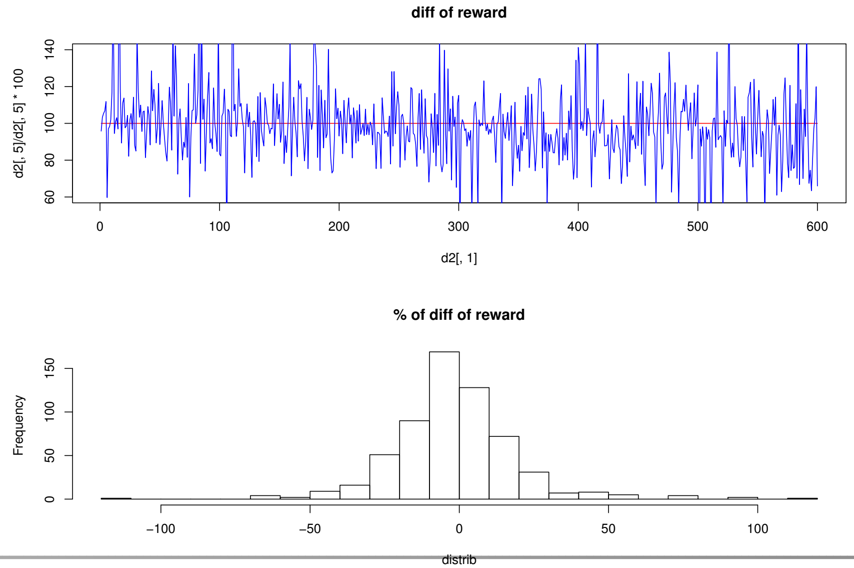
Algorithm

- robots can perceive :
  - other robots
  - prey of type 1
  - prey of type 2
- basic idea :
  - robots discriminate which prey are rewarding
  - they allocate one robot per rewarding prey
  - if a prey is rewarding -> increase proba to take it, else decrease proba
- the net reward is computed by :
  $Reward_{net} = (findTime + retrievalTime) \cdot penalty + reward$

Autonomous behaviour & optimality

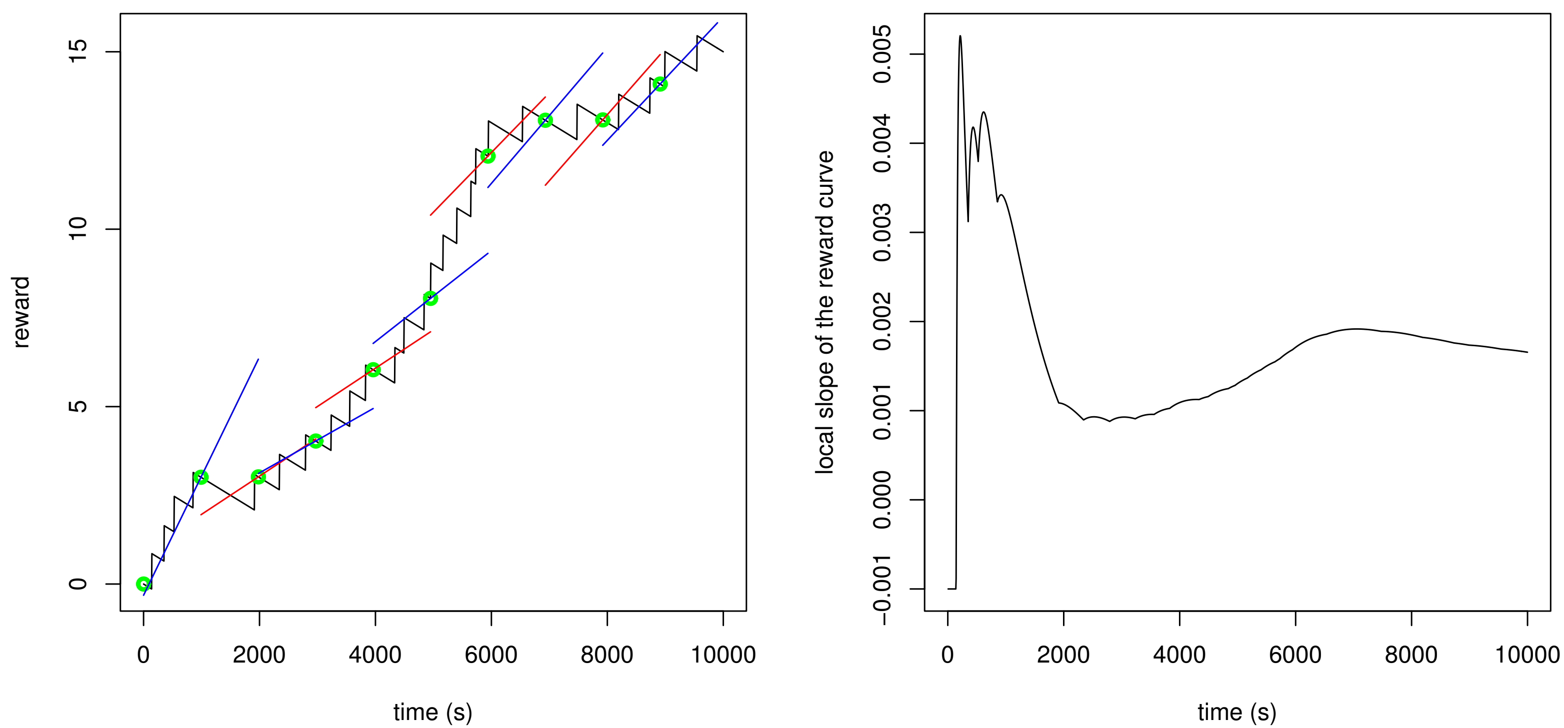
Comparison to optimal predictions

- We measure the predicted optimal reward with respect to the outcome of simulations
- The algorithm is perfectly centered (mean 1%) and the standard error is about 15%



Estimation of the reward rate

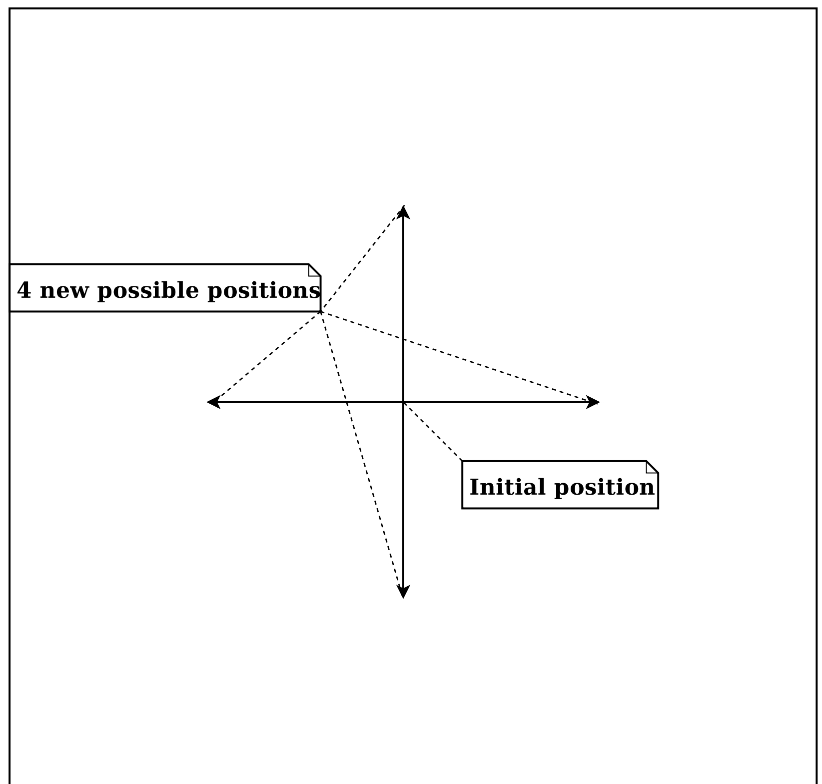
- Robots rely on an weighted regression to estimate the current reward rate.



Autonomous behaviour & optimality

Algorithm of convergence

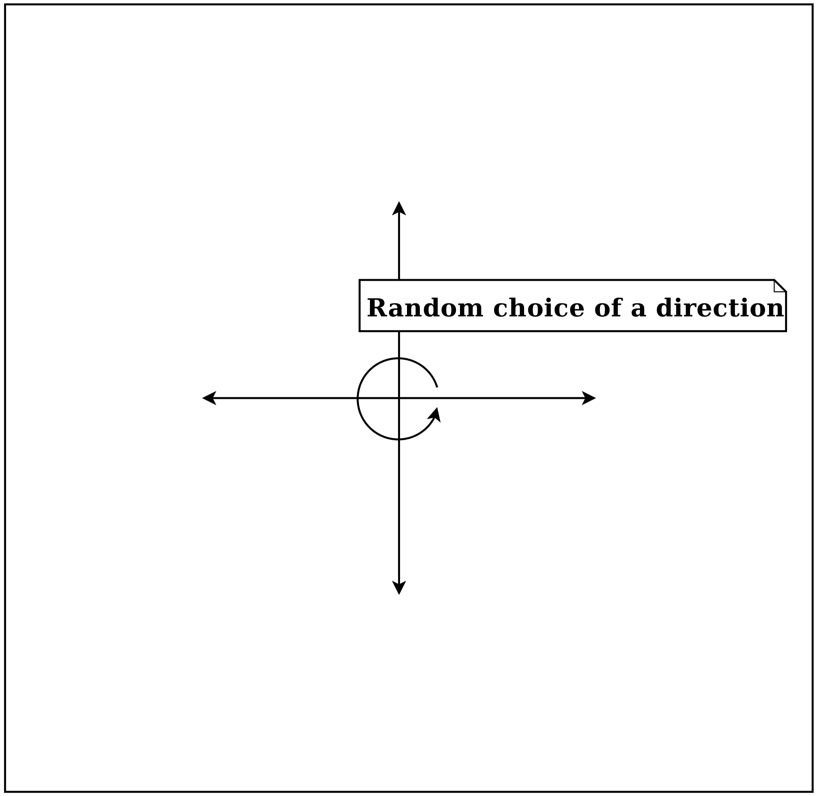
Parameters are a discount factor γ, a step s, a latency l between 2 estimations of the reward rate



Autonomous behaviour & optimality

Algorithm of convergence

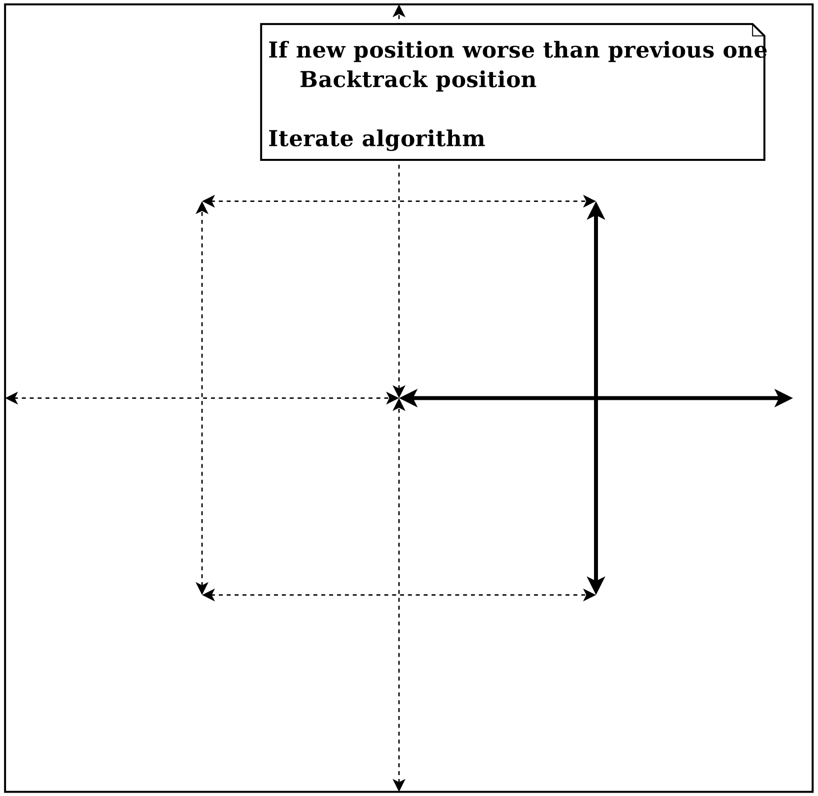
Parameters are a discount factor γ, a step s, a latency l between 2 estimations of the reward rate



Autonomous behaviour & optimality

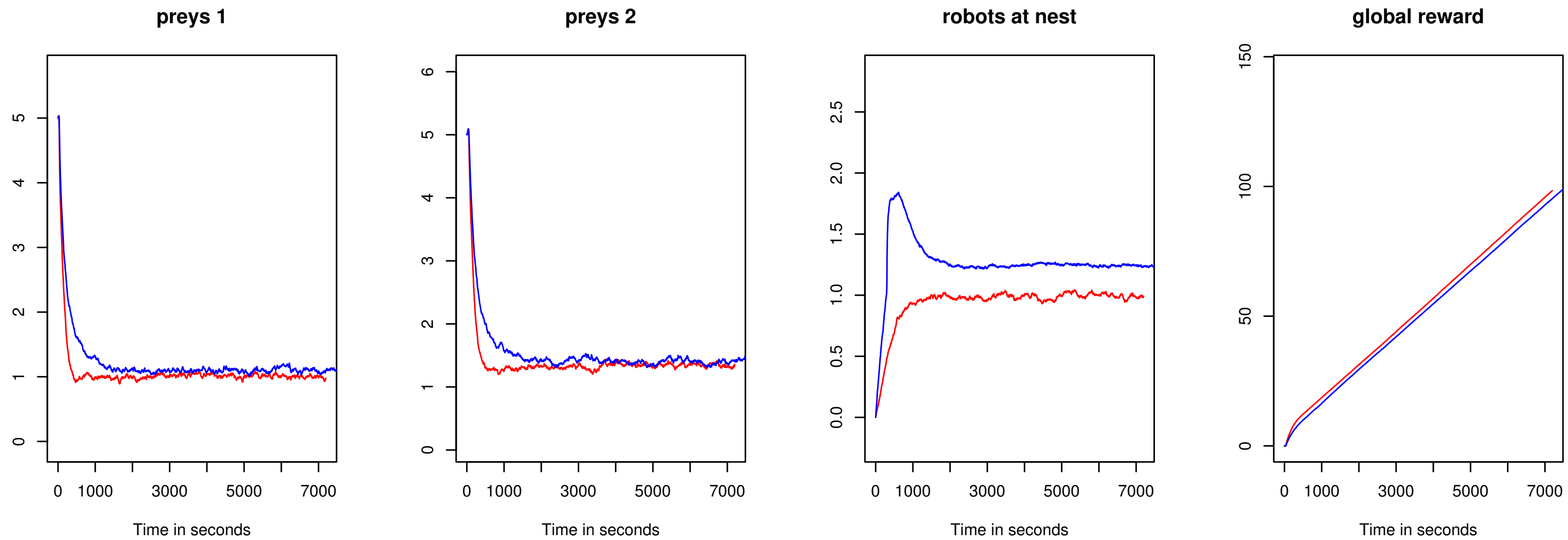
Algorithm of convergence

Parameters are a discount factor γ, a step s, a latency l between 2 estimations of the reward rate



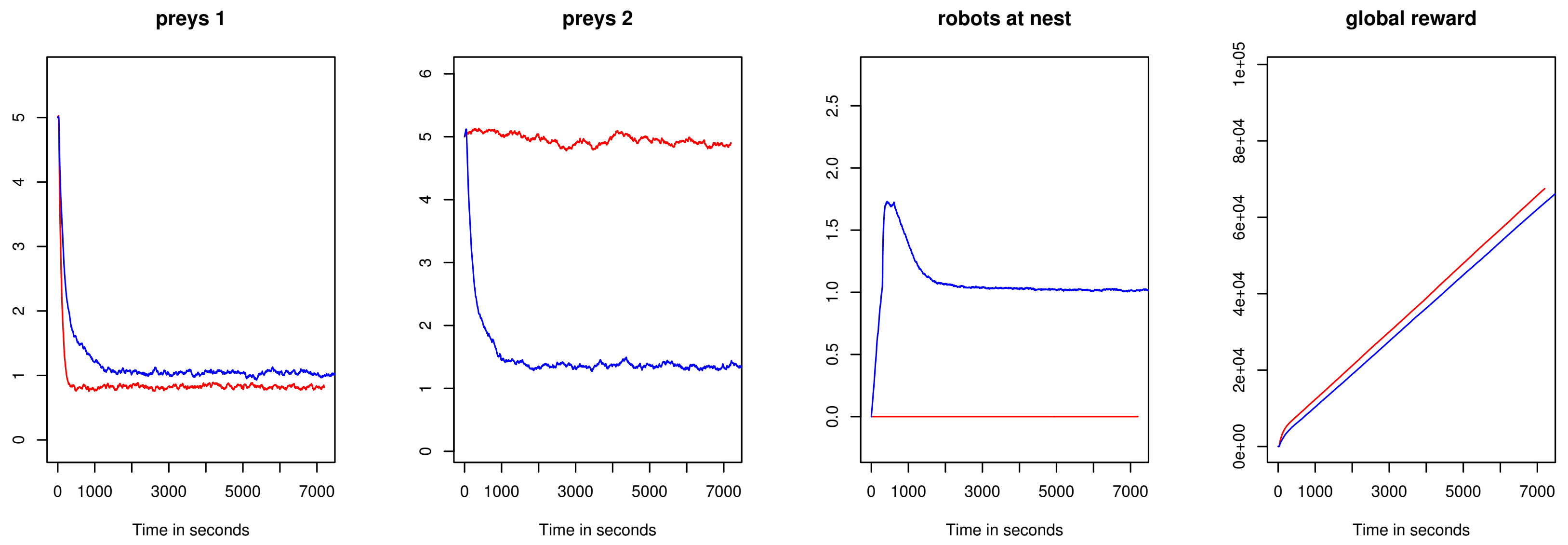
Autonomous behaviour & optimality

Results - case $C_1 = 1, C_2 = 1$



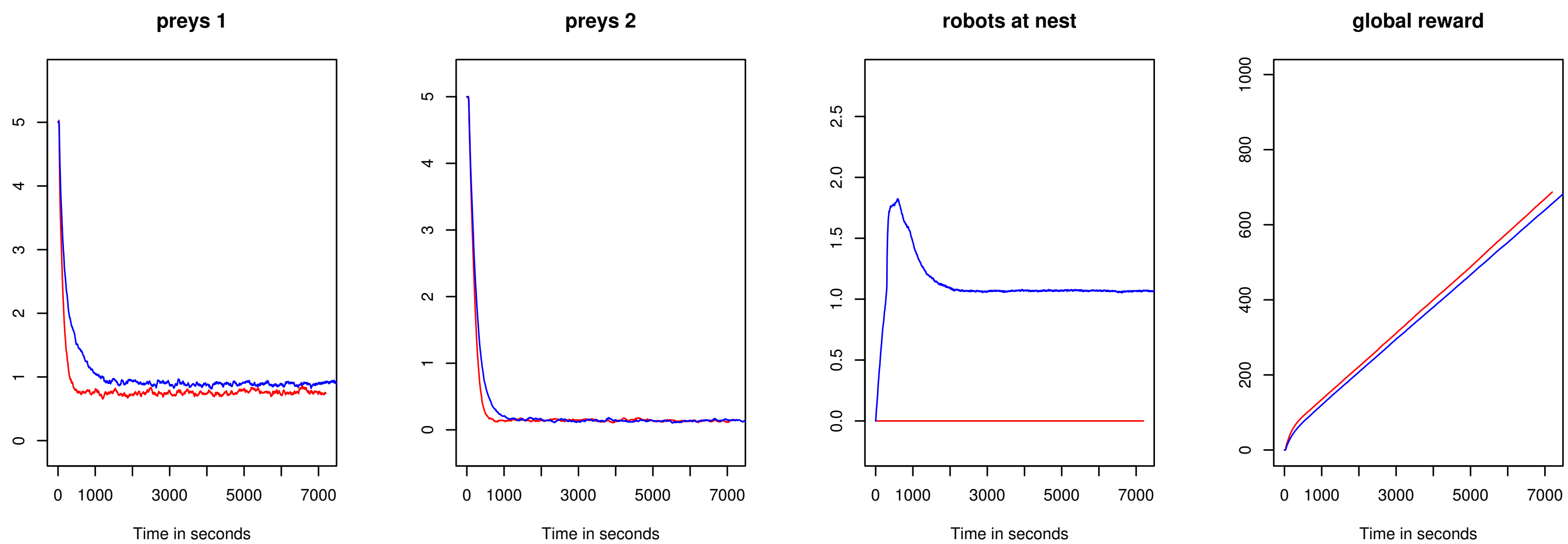
Autonomous behaviour & optimality

Results - case $C_1 = 1, C_2 = 0$



Autonomous behaviour & optimality

Results - case $C_1 = 1, C_2 = X$



Conclusions

- ★ Conclusions
- ★ Future work

Conclusions

- We studied a general problem of task allocation from the point of view of swarm intelligence -> many possible applications
- A model has been devised and validated in a simulated experiment using the prey retrieval paradigm
- An algorithm was proposed that let robots converge fully autonomously toward the best greedy behaviour
- With fixed length experiments a tradeoff between regeneration of the resources and consumption before the end.

Conclusions

Future work

- Simulations results have to be compared against real experiments
- Division of labour will to be studied (is present)
- Adaptivity of the collective will be studied
- Self-regulation of interference effects might be handled

Questions ?

Thank you for your attention !