UNIVERSITY OF Southampton
School of Electronics and Computer Science

# Towards Design Patterns for Robot Swarms

Lenka Pitonakova, Richard Crowder and Seth Bullock

Agents, Interaction and Complexity Group

Institute for Complex Systems Simulation

UNIVERSITY OF Southampton
School of Electronics and Computer Science

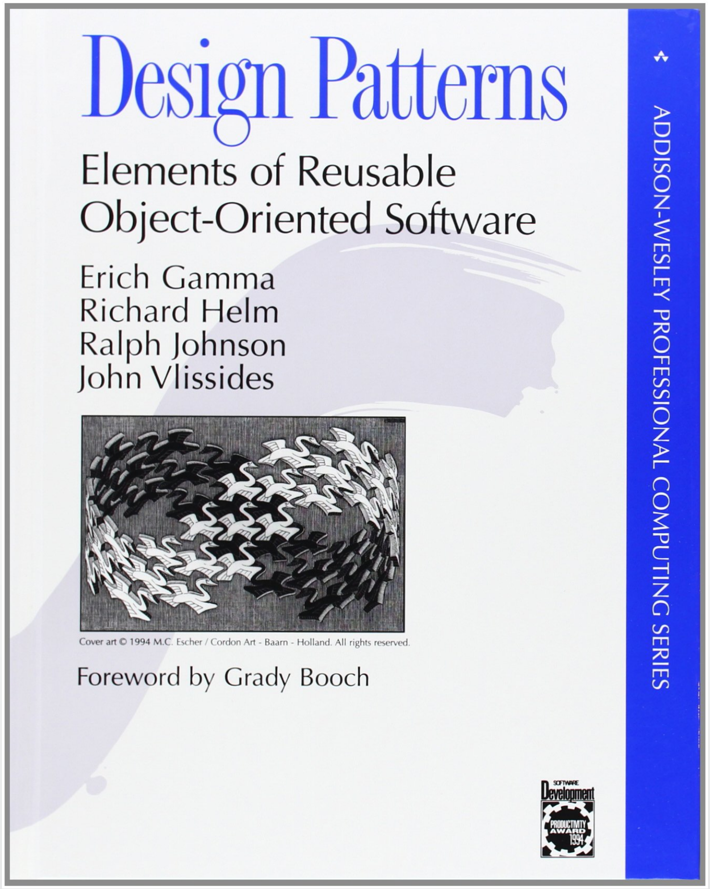
# Towards Design Patterns for Robot Swarms

- What are design patterns
- Our simulation approach
- Swarms in simple environments
- Information value
- Swarms in dynamic environments
- Design pattern principles and example

UNIVERSITY OF Southampton
School of Electronics and Computer Science

# Swarm Robotics

- Currently there is no precise method of selecting robot behaviour for swarms
- In OO software engineering, design patterns help with system design



UNIVERSITY OF Southampton
School of Electronics and Computer Science

# Swarm Robotics

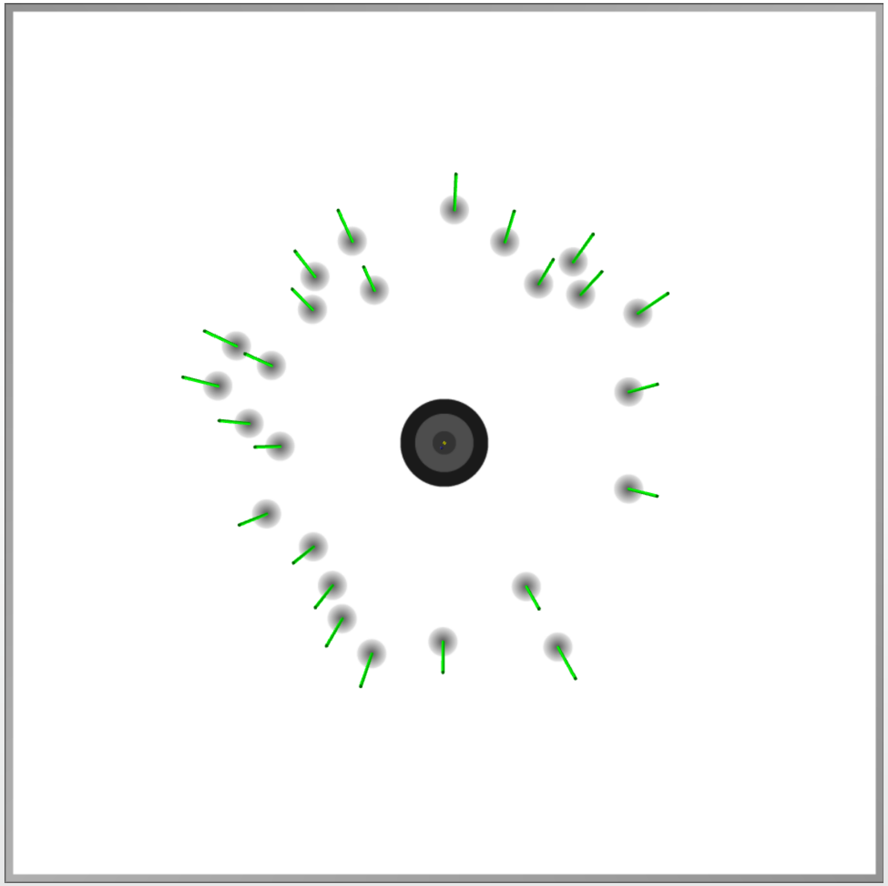
- Swarm robotics could benefit too:
  - Implicit understanding of collective intelligence
  - Modularity of behaviours
  - Mission-specific implementation

UNIVERSITY OF Southampton
School of Electronics and Computer Science

# Our Approach

- 3D simulations with realistic physics
- Parameter sweeps: robots, environment
- Detailed performance analysis
- Information flow analysis
- Information-to-work cost analysis
- Design pattern creation

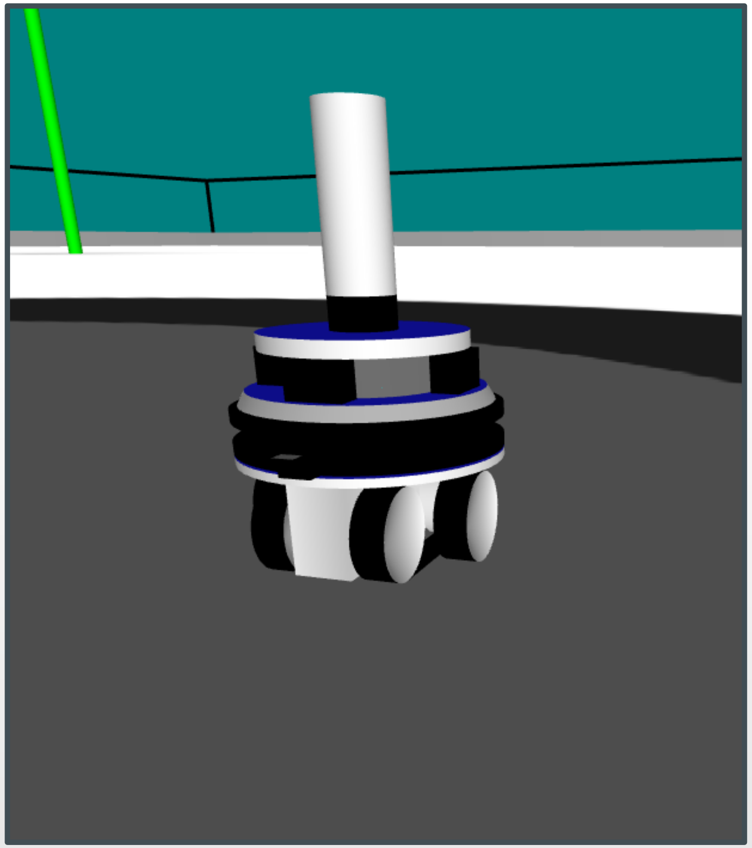
# Simulation work



- Environment variations:
  - Maintenance vs Foraging
  - Different number of tasks
    - $V_T = 100 / N_T$
  - Tasks of different utilities
  - Dynamic tasks

# Simulation work

- Swarm types:
  - Solitary: no communication
  - Local Broadcasters: recruit near tasks
  - Bee Swarms: recruit in the base
- Swarm parameters:
  - Behaviour-specific
  - Different number of robots



# Example: Simple environments

- Static tasks of the same utility
- Which swarm obtains all the reward the fastest?
- More tasks
  - -> smaller task return + easier to find
    - -> solitary foraging favoured

- Less tasks
  - -> bigger task return but harder to find
    - -> recruitment favoured

UNIVERSITY OF Southampton
School of Electronics and Computer Science

# Example: Simple environments

Completion time, Solitary robots vs. Local broadcasters vs. Bee swarm

## Maintenance, 25 robots

| Winners | Num of tasks | Max distance | Task reward |
|---|---|---|---|
| | 25 | 5 | 4 |
| | 25 | 9 | 4 |
| | 25 | 13 | 4 |
| | 25 | 17 | 4 |
| | 25 | 21 | 4 |
| | 4 | 5 | 25 |
| | 4 | 9 | 25 |
| | 4 | 13 | 25 |
| | 4 | 17 | 25 |
| | 4 | 21 | 25 |
| | 2 | 5 | 50 |
| | 2 | 9 | 50 |
| | 2 | 13 | 50 |
| | 2 | 17 | 50 |
| | 2 | 21 | 50 |
| | 1 | 5 | 100 |
| | 1 | 9 | 100 |
| | 1 | 13 | 100 |
| | 1 | 17 | 100 |
| | 1 | 21 | 100 |

## Foraging, 25 robots

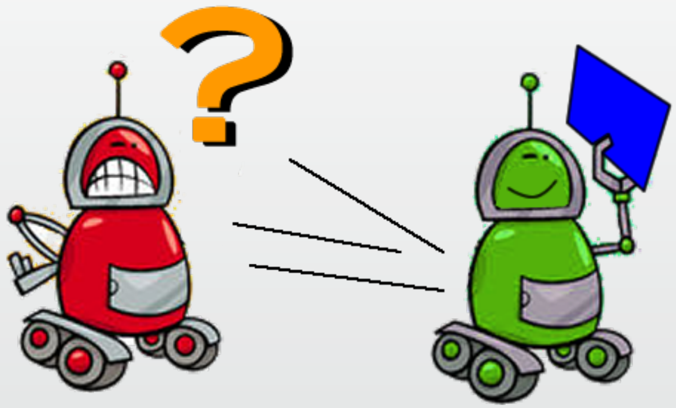
| Winners | Num of tasks | Max distance | Task reward |
|---|---|---|---|
| | 25 | 5 | 4 |
| | 25 | 9 | 4 |
| | 25 | 13 | 4 |
| | 25 | 17 | 4 |
| | 25 | 21 | 4 |
| | 4 | 5 | 25 |
| | 4 | 9 | 25 |
| | 4 | 13 | 25 |
| | 4 | 17 | 25 |
| [too difficult] | 4 | 21 | 25 |
| | 2 | 5 | 50 |
| | 2 | 9 | 50 |
| | 2 | 13 | 50 |
| | 2 | 17 | 50 |
| [too difficult] | 2 | 21 | 50 |
| | 1 | 5 | 100 |
| | 1 | 9 | 100 |
| | 1 | 13 | 100 |
| | 1 | 17 | 100 |
| [too difficult] | 1 | 21 | 100 |

UNIVERSITY OF Southampton
School of Electronics and Computer Science

# Example: Simple environments

- Robot-robot interference:
  - Physical

  - Environmental



# Example: Simple environments

- More robots
  - -> more communication
    - -> communication effects (good and bad!) more pronounced
    - -> winning strategies more environment-specific

UNIVERSITY OF Southampton

School of Electronics and Computer Science

# Example: Simple environments

Completion time, Solitary robots vs. Local broadcasters vs. Bee swarm

## Foraging, 10 robots

| Winners | | | Num of tasks | Max distance | Task reward |
|---|---|---|---|---|---|
| | | | 25 | 5 | 4 |
| | | | 25 | 9 | 4 |
| | | | 25 | 13 | 4 |
| | | | 25 | 17 | 4 |
| | | | 25 | 21 | 4 |
| | | | 4 | 5 | 25 |
| | | | 4 | 9 | 25 |
| | | | 4 | 13 | 25 |
| | | | 4 | 17 | 25 |
| | | | 4 | 21 | 25 |
| | | | 2 | 5 | 50 |
| | | | 2 | 9 | 50 |
| | | | 2 | 13 | 50 |
| | | | 2 | 17 | 50 |
| | | | 2 | 21 | 50 |
| | | | 1 | 5 | 100 |
| | | | 1 | 9 | 100 |
| | | | 1 | 13 | 100 |
| | | | 1 | 17 | 100 |
| | | | 1 | 21 | 100 |

## Foraging, 50 robots

| Winners | | | Num of tasks | Max distance | Task reward |
|---|---|---|---|---|---|
| | | | 25 | 5 | 4 |
| | | | 25 | 9 | 4 |
| | | | 25 | 13 | 4 |
| | | | 25 | 17 | 4 |
| | | | 25 | 21 | 4 |
| | | | 4 | 5 | 25 |
| | | | 4 | 9 | 25 |
| | | | 4 | 13 | 25 |
| | | | 4 | 17 | 25 |
| | | | 4 | 21 | 25 |
| | | | 2 | 5 | 50 |
| | | | 2 | 9 | 50 |
| | | | 2 | 13 | 50 |
| | | | 2 | 17 | 50 |
| | | | 2 | 21 | 50 |
| | | | 1 | 5 | 100 |
| | | | 1 | 9 | 100 |
| | | | 1 | 13 | 100 |
| | | | 1 | 17 | 100 |
| | | | 1 | 21 | 100 |

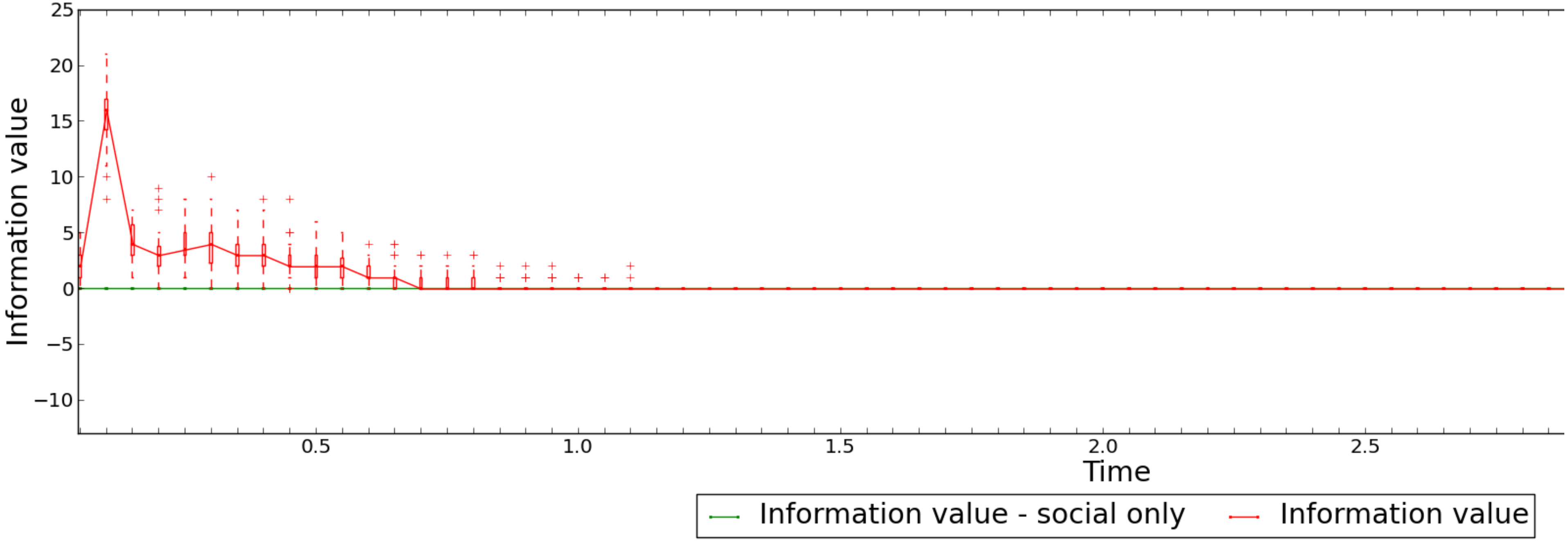
# Information value

- What is the value of new information for a robot?
  - Reward that can be extracted from a task per volume unit, compared to a reward the robot would receive using some old information

- $I = U_{new} – U_{old}$
- For scouts and unemployed robots, $I = U_{new}$
- For recruited robots, it can be positive or negative

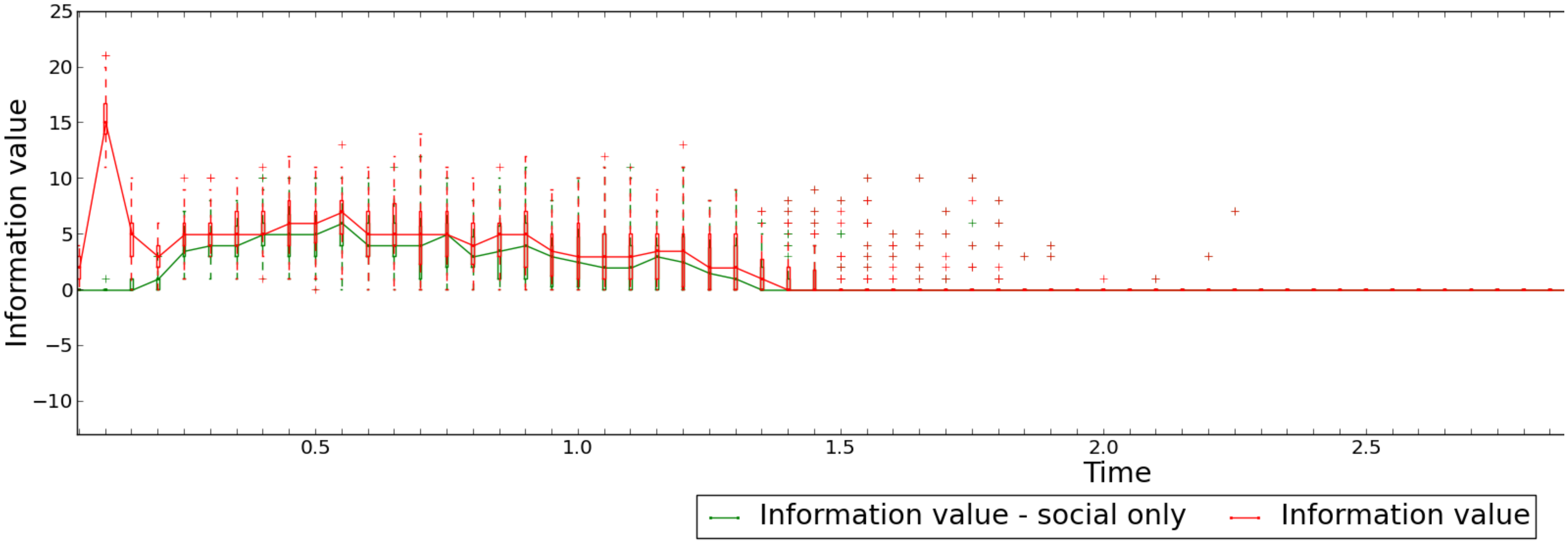
# Information value

UNIVERSITY OF Southampton
School of Electronics and Computer Science




Foraging, 25 deposits, 25 solitary robots

# Information value

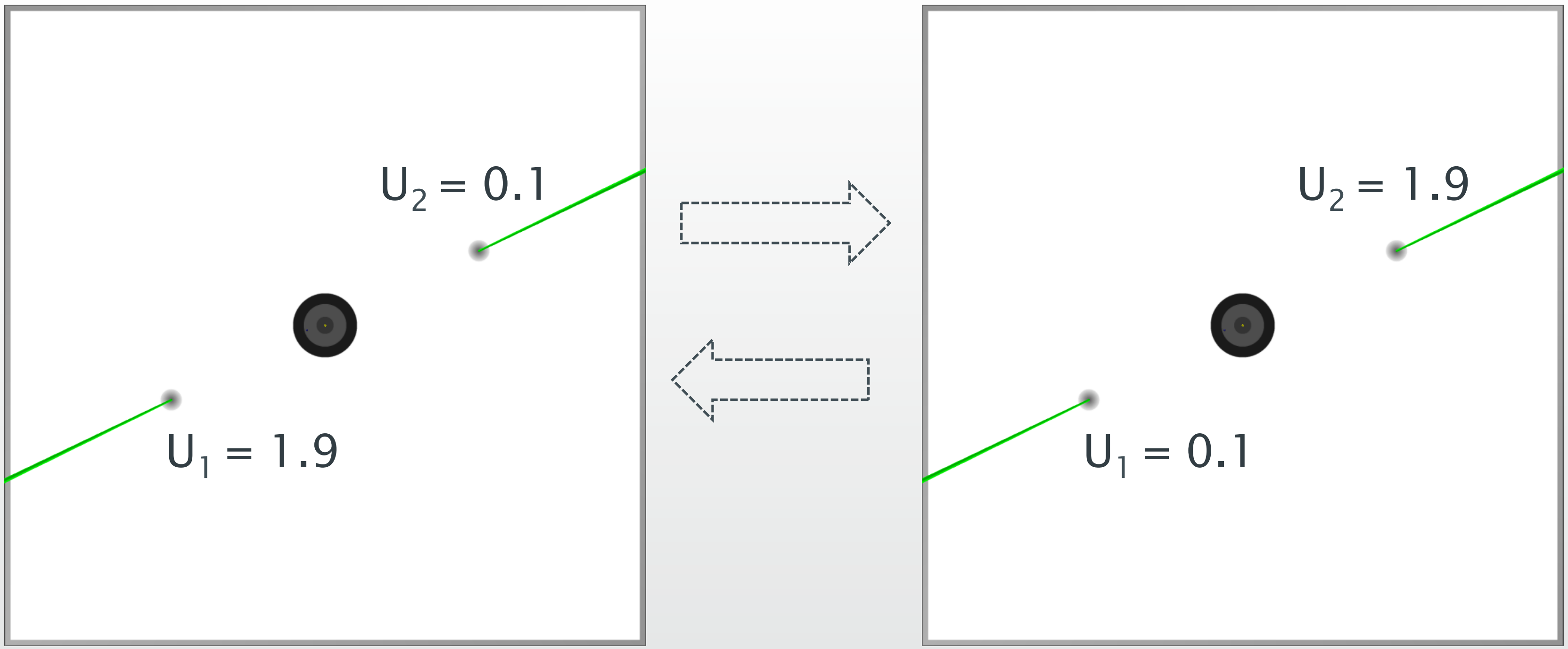



Foraging, 25 deposits, 25 bee swarm robots

# Example: Dynamic task utilities

UNIVERSITY OF Southampton
School of Electronics and Computer Science

- Bee swarm able to choose between tasks of different utilities to maximise reward during foraging

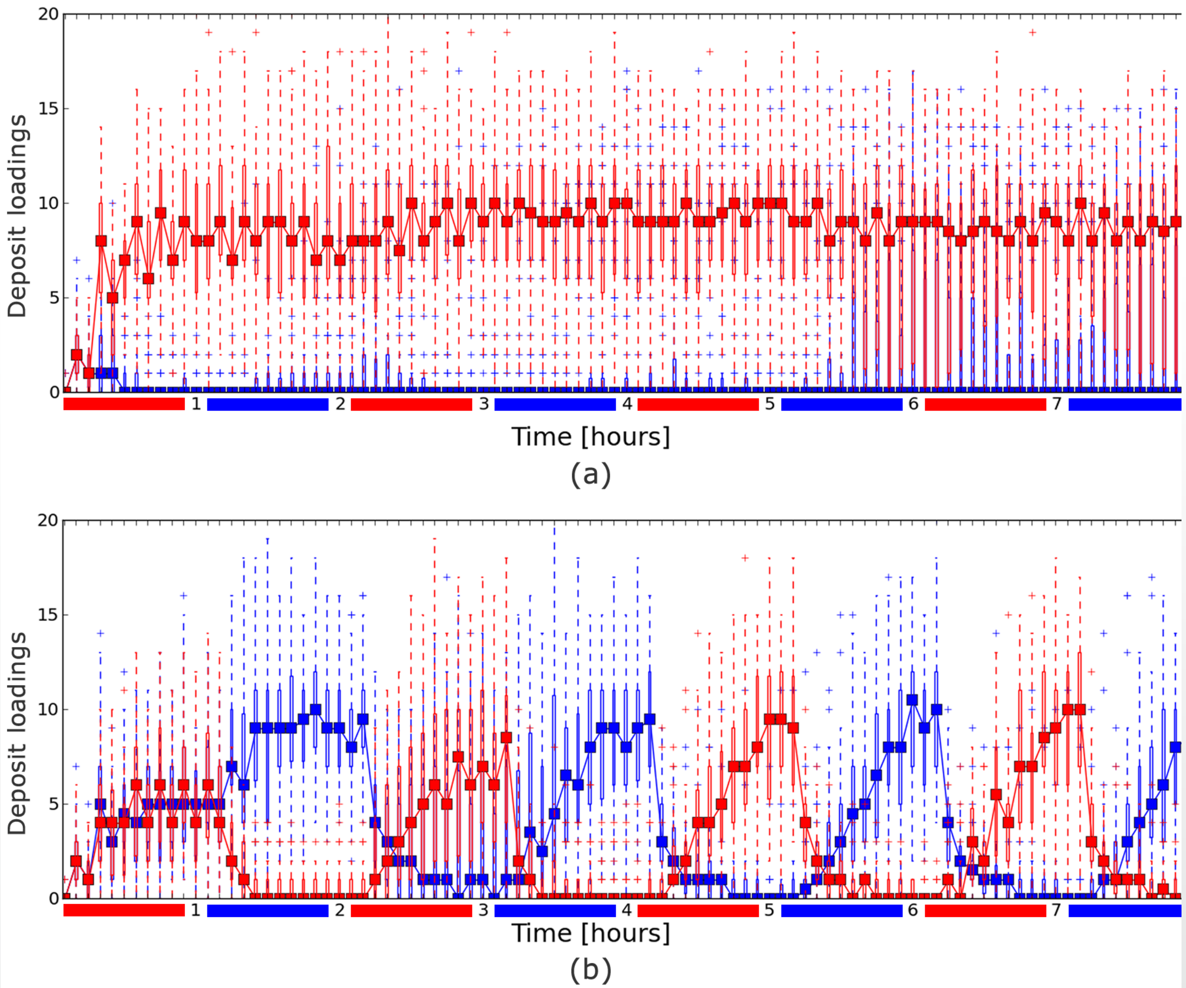


# Example: Dynamic task utilities

- EE = u/d
- ‘Beggers’
  - Robots in the base compare EE of their own tasks to that of other robots and can switch
  - Deplete the best task quickly, then move to another
- ‘Checkers’
  - Robots abandon a task if its EE dropped
  - Spread across tasks more evenly
  - Faster response to environmental change

UNIVERSITY OF Southampton
School of Electronics and Computer Science

# Example: Dynamic task utilities



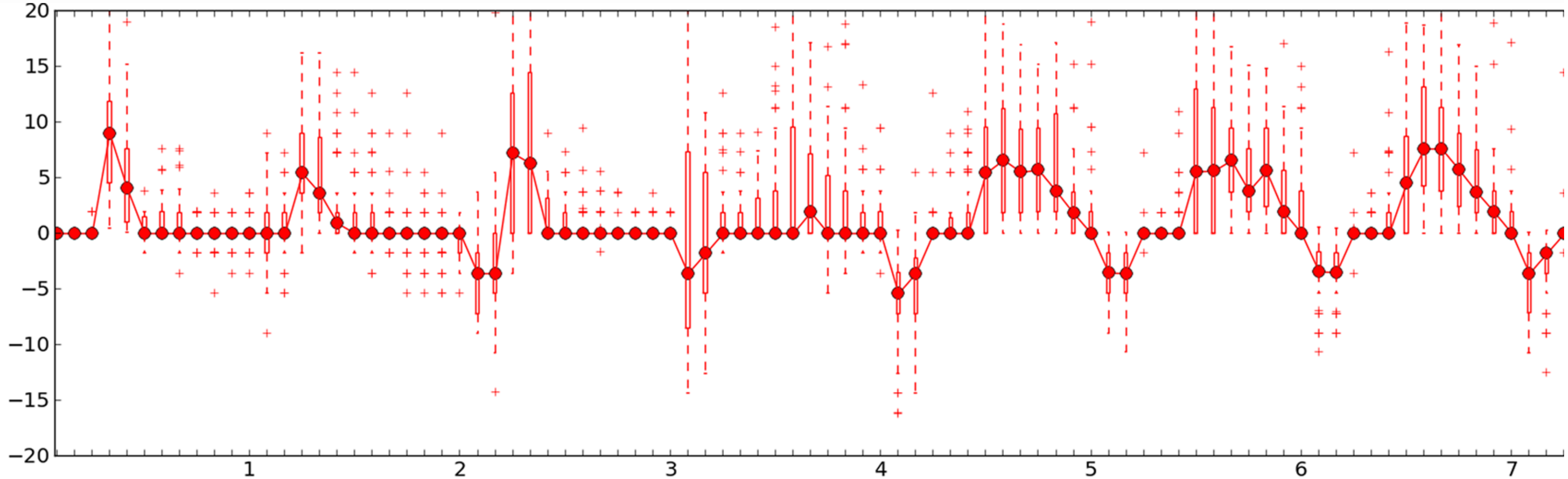

2 nearby deposits with changing utility, number of loadings from deposits

(a) 25 Beggers
(b) 25 Checkers

UNIVERSITY OF Southampton
School of Electronics and Computer Science

# Example: Dynamic task utilities

- Based on information value, we can identify swarm work modes

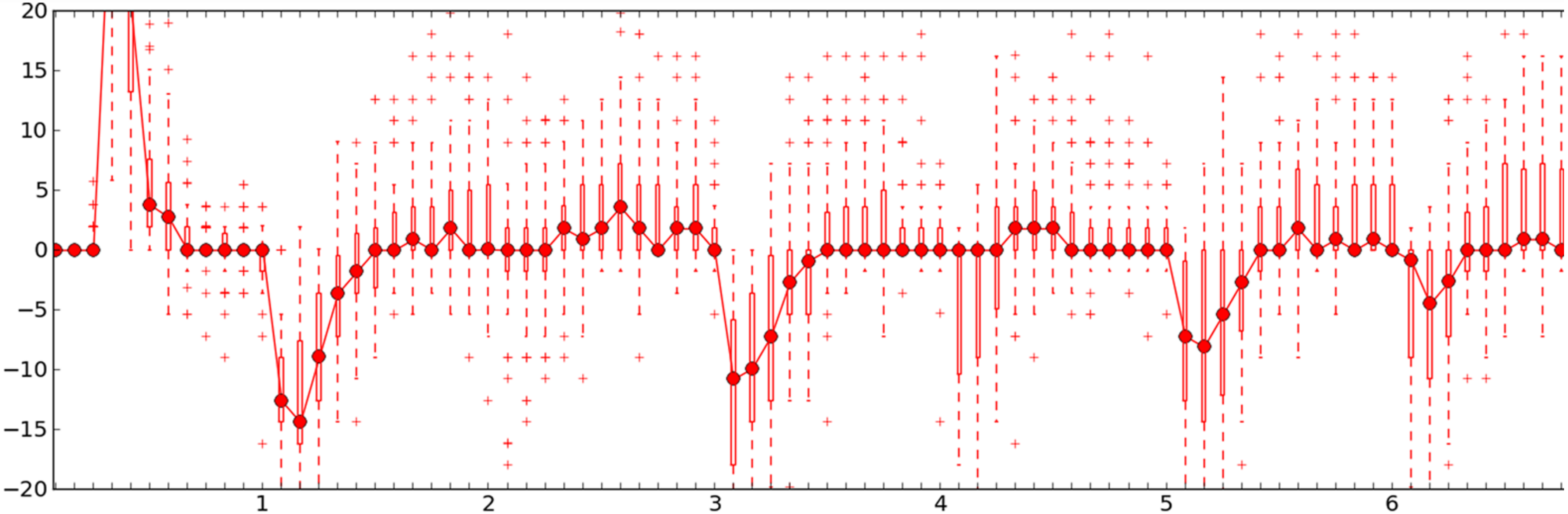



Switching (Long-range checkers, 2 deposits, 25 robots, D=9m)

UNIVERSITY OF Southampton
School of Electronics and Computer Science

# Example: Dynamic task utilities

- Based on information value, we can identify swarm work modes

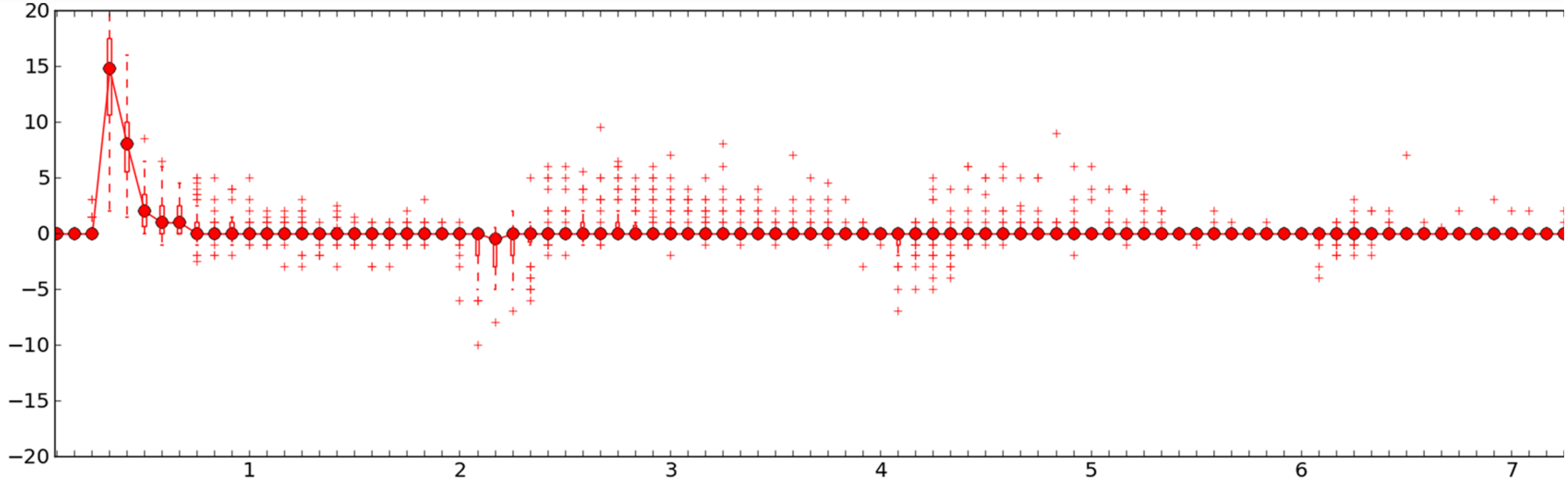



Delayed switching (Long-range beggers, 2 deposits, 45 robots, D=9m)

# Example: Dynamic task utilities

UNIVERSITY OF Southampton
School of Electronics and Computer Science

- Based on information value, we can identify swarm work modes

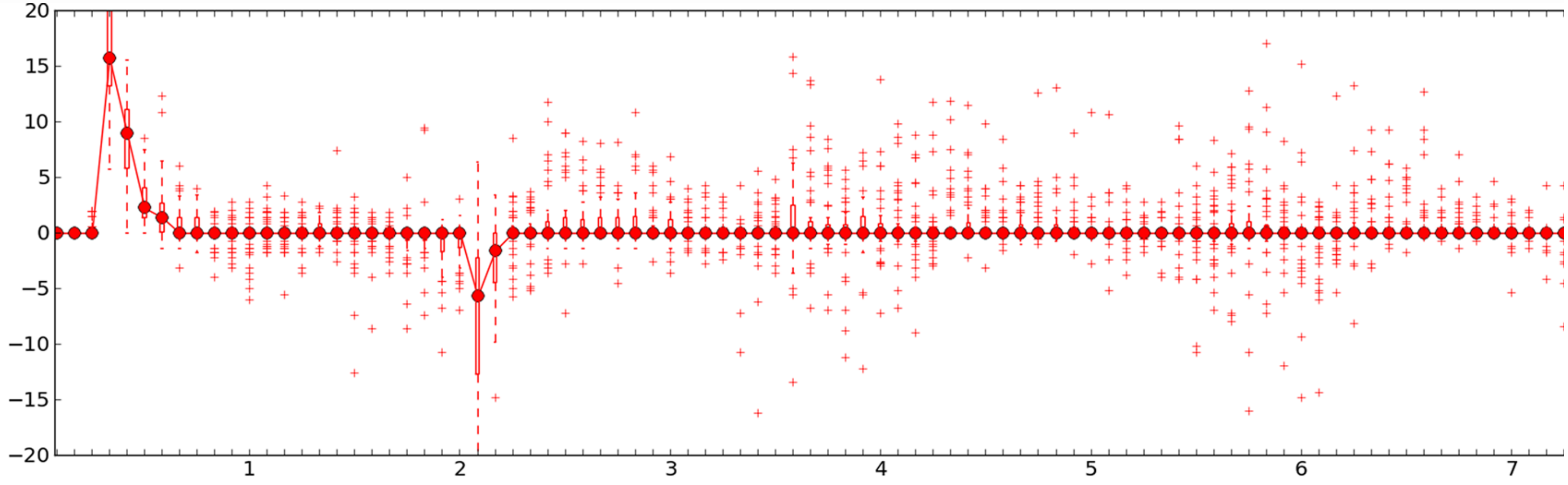



Locked (Long-range beggers, 2 deposits, 25 robots, D=9m)

# Example: Dynamic task utilities

UNIVERSITY OF Southampton
School of Electronics and Computer Science

- Based on information value, we can identify swarm work modes




Indecisive (Long-range beggers, 4 deposits, 25 robots, D=9m)

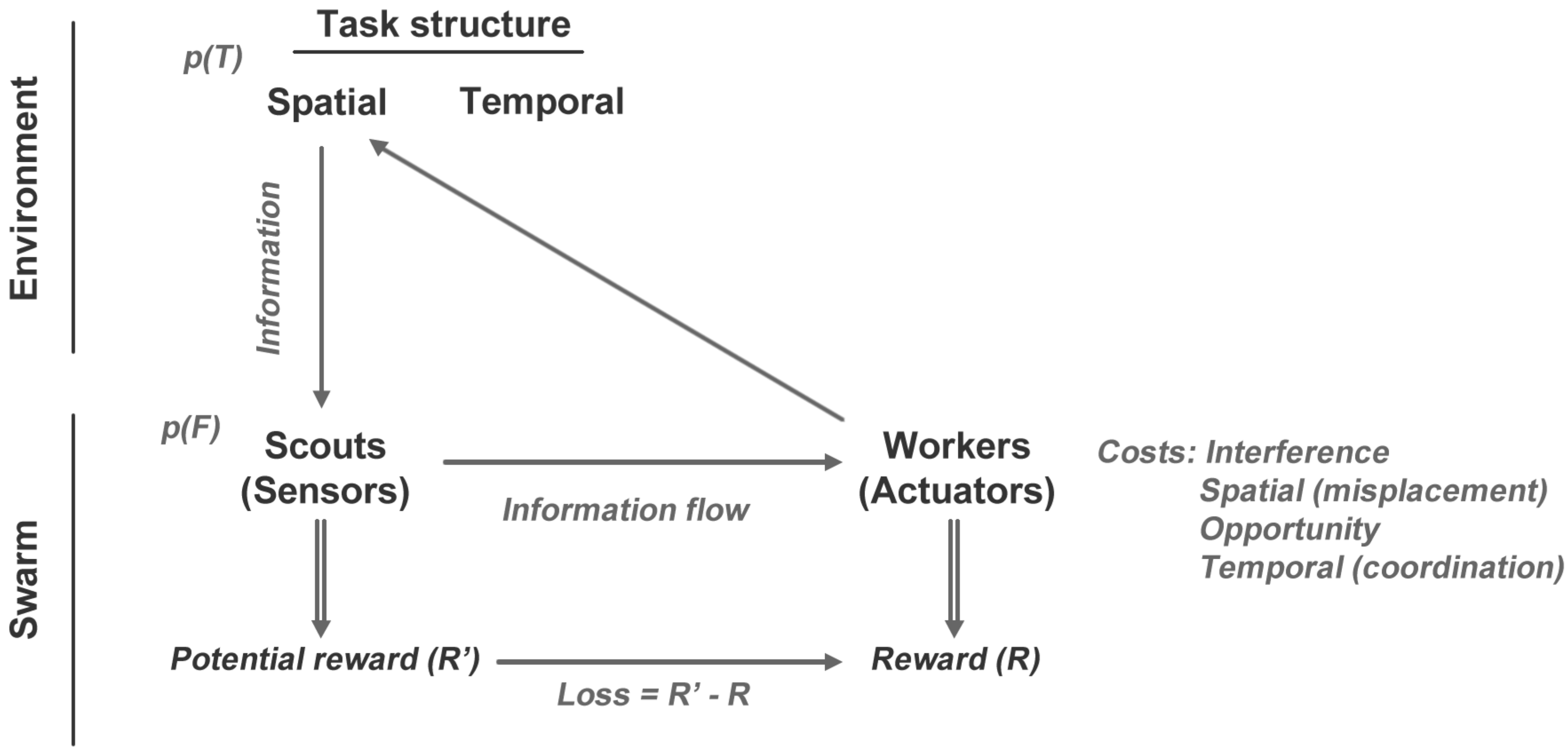
UNIVERSITY OF Southampton
School of Electronics and Computer Science

# Information value: General findings

- The correct information flow, given a particular environment, promotes swarm plasticity

- Across a range of environments, different swarm types have different likelihoods to exhibit plasticity
  - Faster information flow -> better performance but likelihood of plasticity is sacrificed

# The perception-action loop



UNIVERSITY OF Southampton
School of Electronics and Computer Science

# The perception-action loop

- How can we match swarm behaviour to environment in order to minimise costs and maximise work?

- Design patterns

# Design patterns

- Modular
  - How to navigate an unknown environment
  - Where and when is information transferred
  - How does old information gets updated

# Design patterns

- Specified by
  - Unambiguous name
  - Problem
  - Solution
  - Parameters
  - Trade-offs
  - (Effects when combined with other patterns)

# Information exchange centre

- Problem: how to let other robots know about tasks?
- Solution: define a meeting place where robots can exchange information. Unsuccessful scouts come to this place to meet with successful scouts.

- Parameters:
  - Scouting time
  - ‘Recruitment’ time

# Information exchange centre

- Trade-offs
  - Promotes spatio-temporal synchronisation of robot work (good when tasks are hard to find, or for cooperative tasks)
  - Potential high cost of traveling to tasks if the IEC is far away (not suitable for maintenance missions)
  - Low values of scouting time and high values of recruitment time cause poor exploration of the environment (hard to calibrate for environments of unknown size)

# Current work..

- Additional experiments
  - Stigmergy-based recruitment (ants)
  - Dynamic environments
  - Tasks that require cooperation
- Visualisation of relationships between robot states, data and the environment
- Creation and classification of design patterns

# Thank you.
# Questions?

UNIVERSITY OF Southampton
School of Electronics and Computer Science