University of Hertfordshire

Approximation of Empowerment in the Continuous Domain

C. Salge, C. Glackin & D. Polani

University of Hertfordshire

CORBYS

Cognitive Control Framework for Robotic Systems

SEPIA UNIT

Approximation of Empowerment in the Continuous Domain

- What is *Empowerment?*
- Why go into the *Continuous Domain?*
- How to *Approximate?*

- Application?

CORBYS
Cognitive Control Framework for Robotic Systems

Approximation of Empowerment in the Continuous Domain

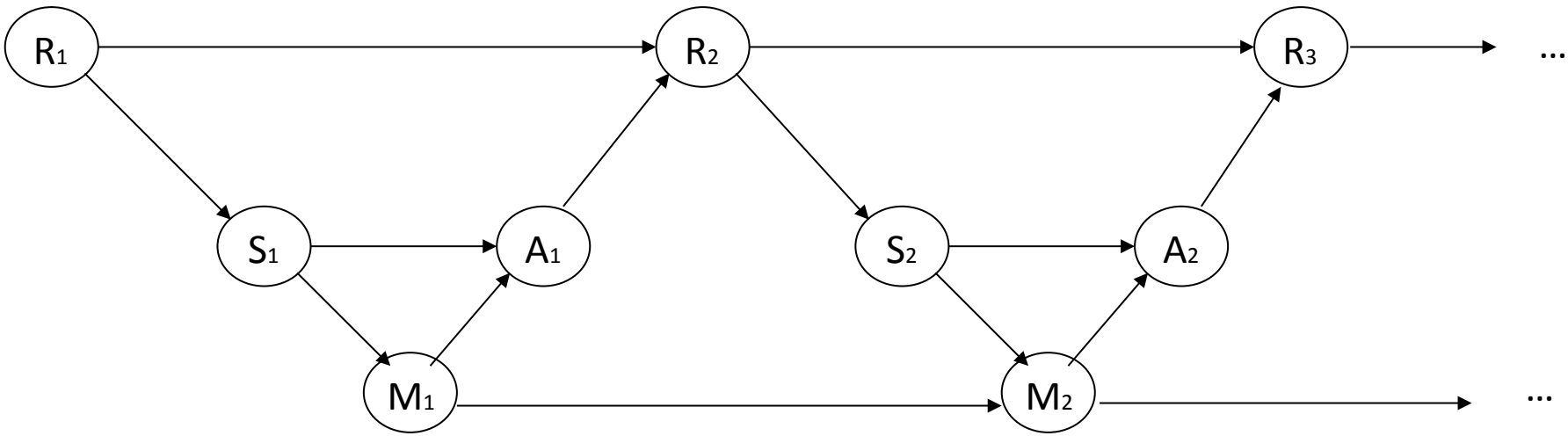
- What is *Empowerment*?
- Why go into the *Continuous Domain*?
- How to *Approximate*?

- Application?

CORBYS
Cognitive Control Framework for Robotic Systems

A Key Problem of Guided Self-Organisation

You are an agent with Sensor Input ***S*** and Actuator Output ***A*** in an Environment ***R***:



How to act?

Are there preferred states?

CORBYS
Cognitive Control Framework for Robotic Systems

More Precisely:

- Define a utility function for each state of r, which depends only on the interaction of the agent with the environment and the dynamics in between actuators and sensor.
- know world model -> no learning
- Not an action policy, but can be used to generate one, i.e. by using greedy maximisation of empowerment

Act always so as to increase the number of choices.

Handle stets so, daß die Anzahl der Wahlmöglichkeiten größer wird.

Heinz von Foerster

What about choice?

Ideally, an agent has a maximal number of different action choices, which predictably lead to different exclusive and observable outcomes.

Bad cases:

-to few choices

-actions lead to random outcomes

-actions lead to the same outcomes

-actions only affect states that cannot be perceived.

Empowerment Formalism

- For an agent with actuator output **A** and sensor input **S** this is captured by the channel capacity for the state **r**:

$$C(r) = \max_{p(a)} I(A; S|r)$$

  - r: state of the world
  - p(a): capacity achieving distribution

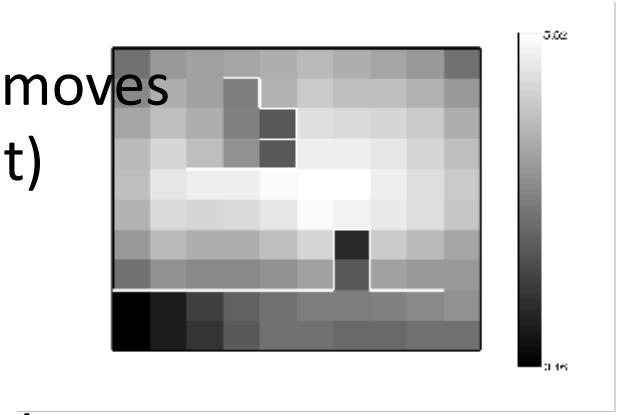
CORBYS
Cognitive Control Framework for Robotic Systems

Alexander Klyubin et. al.

Maze Example

*Actuator Output **A**:*

5-step sequence of moves (up, down, left, right)



*Sensor Input **S**:*

position in maze at the end of the actions sequence

White lines are impassable walls.

CORBYS
Cognitive Control Framework for Robotic Systems

Alexander Klyubin et. al.

Collective Behaviour

Scenario:

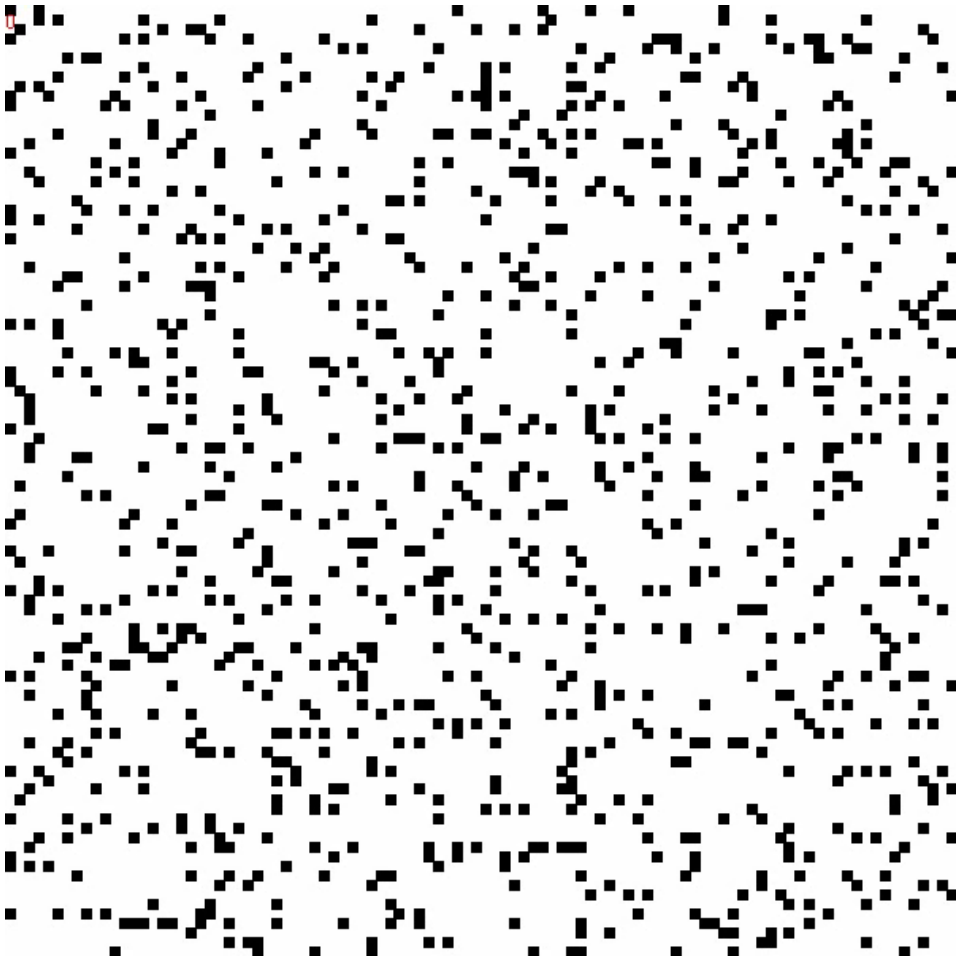
- Large grid world with several agents
- Actions: 3-step action sequence
  - Actions are *up, down, left, right*
- Sensor Input: Agent concentration in the four quadrants around the agent.

Collective Behaviour

Adaptation:

- All agent in a population act according to the same strategy.
- A strategy defines for every sensor input what action an agent should take
- Strategies are encoded in a genome and then evolved.
- Fitness is defined as the average empowerment of the agents in the population.

Collective Behaviour



CORBYS
Cognitive Control Framework for Robotic Systems

Philip Capdepuy et. al.

Approximation of Empowerment in the Continuous Domain

- What is *Empowerment*?
- Why go into the *Continuous Domain*?
- How to *Approximate*?

- Application?

CORBYS
Cognitive Control Framework for Robotic Systems

Why the Continuous Domain?

- Robots:
  - Sensor Readings are often real numbers
  - Actuation is often a choice from a continuous interval at discrete points in time.

Continuous Empowerment

- One possible solution:
  - Discretization via binning of actions and Monte Carlo Integration over Outcomes assume to be Multivariate Gaussians. [Tobias Jung et. al.]

$$C(x) := \max_{p(\vec{a})} \sum_{v=1}^{N_n} p(\vec{a}_v) \int_X p(x' \mid x, \vec{a}_v).\log\left\{ \frac{p(x' \mid x, \vec{a}_v)}{\sum_{i=1}^{N_n} p(x' \mid x, \vec{a}_i) p(\vec{a}_i)} \right\} dx'$$

- Problems:
  - Large amount of discrete states make computation very demanding.

Approximation of Empowerment in the Continuous Domain

- What is *Empowerment*?
- Why go into the *Continuous Domain*?
- How to *Approximate*?

- Application?

CORBYS
Cognitive Control Framework for Robotic Systems

Model as:
MIMO Linear Gaussian Channel

- S: Vector of Continuous Outputs
- A: Vector of Continuous Inputs
- T: Transformation Matrix
- Z: Isotropic Gaussian Noise $\sim\mathcal{N}(0,1)$
- Power Limitation: $E[A^2] < P$

$$S = TA + Z$$

Linear Gaussian Channel

- S: Continuous Output
- A: Continuous Input
- c: constant
- Z: Isotropic Gaussian Noise $\sim\mathcal{N}(0,1)$
- Power Limitation: $E[A^2] < P$

$$S = cA + Z$$

- Channel capacity achieving solution is Gaussian distribution of Input:

$$A \sim \mathcal{N}\left(0, \sqrt{P}\right)$$

$$C = \frac{1}{2}\log(1 + cP)$$

Solve for:
MIMO Linear Gaussian Channel

- Use Singular Value Decomposition to decompose Problem to parallel linear Gaussian Channels.
- Allocate Power with Water-filling Algorithm for each spectral channel.

Telatar, E., Capacity of multi-antenna Gaussian channels, 1999

Approximation of Empowerment in the Continuous Domain

- What is *Empowerment?*
- Why go into the *Continuous Domain?*
- How to *Approximate?*

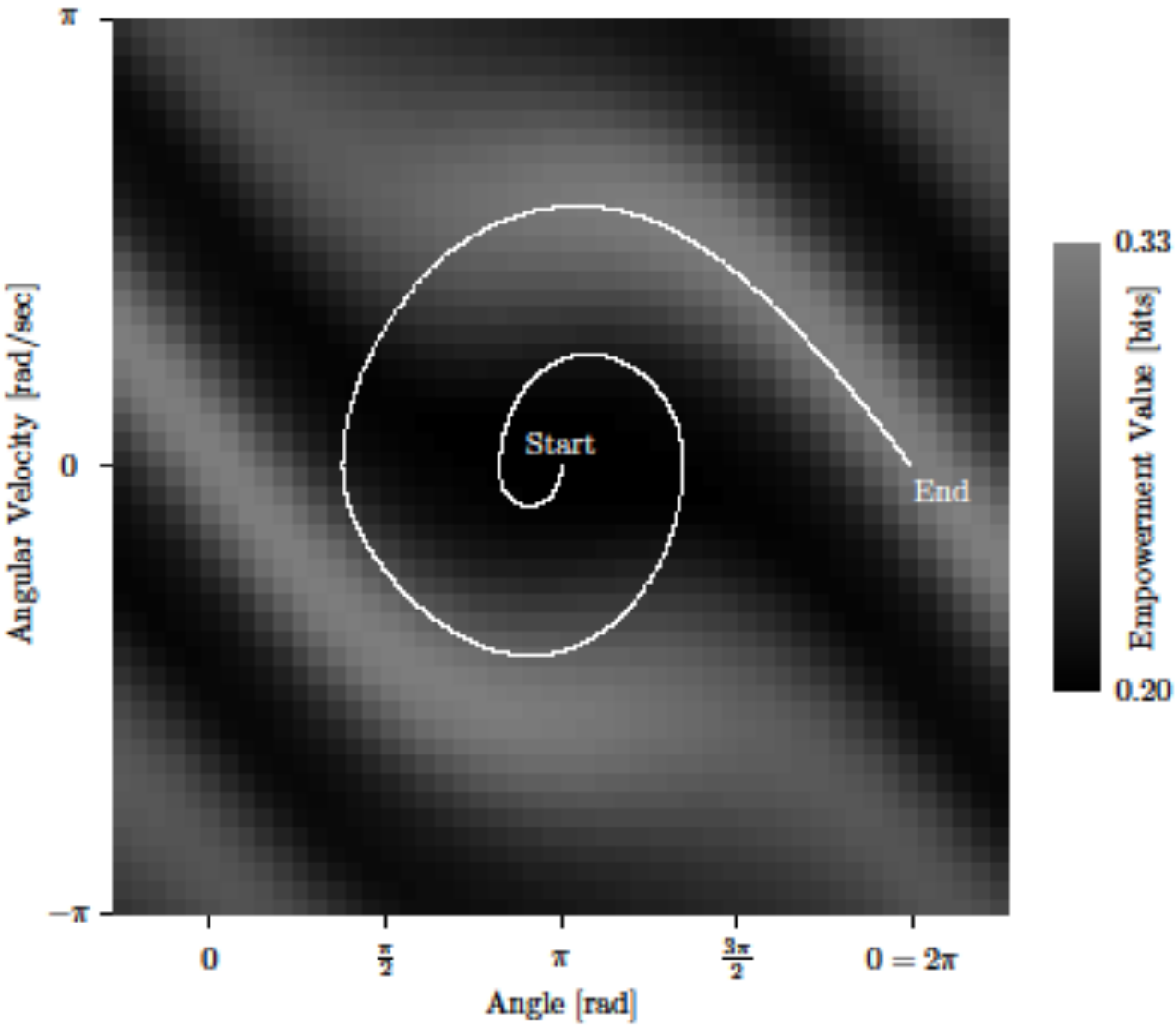
- Application?

CORBYS
Cognitive Control Framework for Robotic Systems

Pendulum

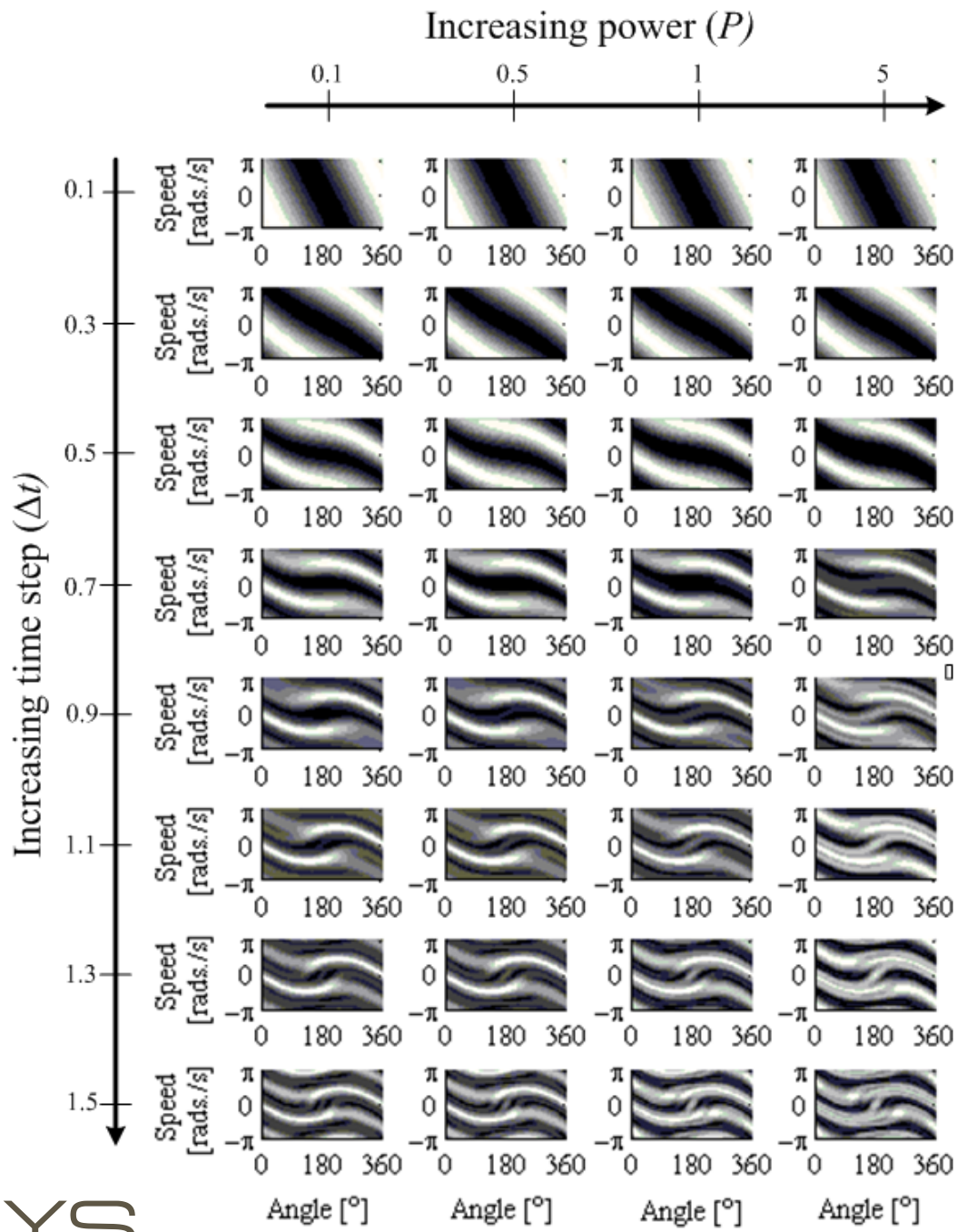
- Simple Pendulum
- Action Input: Setting the acceleration at the beginning of three consecutive time steps with length Δt, which is then maintained for the duration of Δt.
- Sensor Output: The angle and speed of the pendulum after the third time step.

- Parameters:
  - Length of Δt
  - Maximal Acceleration of Pendulum



CORBYS
Cognitive Control Framework for Robotic Systems

Varying Parameters



CORBYS
Cognitive Control Framework for Robotic Systems

Live Demo

Pendulum Live Demo

University of Hertfordshire

Questions ?

Get the paper, now on ACS online:
C. Salge, C. Glackin & D. Polani,
Approximation of Empowerment in the Continuous Domain

CORBYS
Cognitive Control Framework for Robotic Systems

SEPIA UNIT

Bonus Content

What about learning?

How do you get the model?

a.) Analytically derive T from pendulum equations

- Linear Regression on Simulation Data
  - b.) with regular action sequences
  - c.) with random action sequences