Advances in Complex Systems

APPROXIMATION OF EMPOWERMENT IN THE CONTINUOUS DOMAIN

CHRISTOPH SALGE
Adaptive Systems Research Group, University of Hertfordshire, College Lane, Hatfield, AL10 9AB, UK
c.salge@herts.ac.uk

CORNELIUS GLACKIN
Adaptive Systems Research Group, University of Hertfordshire, College Lane, Hatfield, AL10 9AB, UK
c.glackin2@herts.ac.uk

DANIEL POLANI
Adaptive Systems Research Group, University of Hertfordshire, College Lane, Hatfield, AL10 9AB, UK
d.polani@herts.ac.uk

Received (received date)
Revised (revised date)

The *empowerment* formalism offers a goal-independent utility function fully derived from an agent's embodiment. It produces intrinsic motivations which can be used to generate self-organizing behaviours in agents. One obstacle to the application of empowerment in more demanding (esp. continuous) domains is that previous ways of calculating empowerment have been very time consuming and only provided a proof-of-concept. In this paper we present a new approach to efficiently approximate empowerment as a parallel, linear, Gaussian channel capacity problem. We use pendulum balancing to demonstrate this new method, and compare it to earlier approximation methods.

Keywords: Empowerment; Information Theory; Channel Capacity; Perception-Action Loop; Self-Organization; Continuous Domain; Control Theory; Goal Independence

> The road leading to a goal does not separate you from the destination; it is essentially a part of it. — **Charles DeLint**

1. Introduction

The traditional approach to decision making or control is to realize what the goals are, and then figure out how to get to them. But how should one act if there are no goals, or if they are yet unknown. This problem of self-motivated behaviour generation has in recent year been approached from numerous directions, which will be detailed in the related work section.

In this paper we will focus on the *empowerment* formalism [18, 19] which offers a solution by assigning to every state an empowerment value, which does not depend on an externally given goal, but rather on the intrinsic dynamical structure of the agent-environment interaction. The idea then is to enter a state where one's own actions matter the most, where they have the greatest impact on the world as the agent perceives it.

The least empowered is the case where every action will lead to the same outcome, or looking at a more general stochastic interpretation, where the outcome distribution is unaffected by the agent's actions. This "worst case" scenario has vanishing empowerment; it is equivalent to agent "death".

The highest empowerment is achieved in the case where all actions lead to distinct (non specific) and unique outcomes. Thus, the selection of a particular action is reflected in what happens to the agent and there are no ineffectual actions.

Between those extreme cases the empowerment is lowered, either by overlapping results (so several actions will lead to the same outcome) or by actions that have different possible outcomes (so taking an action cannot assure what happens).

We illustrate this with a simple introductory gridworld example. Consider an agent located in a maze where no target state has been specified *a priori*. When the target is later revealed the agent should be in the most advantageous position, meaning it can reach any possible target quickly. In this case, it makes sense to position it in a location of maximal "centrality", i.e. minimum average distance to all possible locations. This, again, turns out to be closely related to the spot from which the largest amount of states can be reached in a fixed number of actions (here: action sequences), i.e. with the highest empowerment value. If one was in a maze and it was not yet clear where the goal is, it would be reasonable to go to a spot from which one can reach the most other places in the maze. In Fig. 1 the empowerment for five step long action sequences is visualized. The central positions, those that have a low average distance to all other positions, are shown to have high empowerment.

Empowerment is formalized, which will be detailed later, as the channel capacity between an agent's actions, and its sensors. Note that this means it can be computed from the agent's perspective, since it only requires access to the agent's sensors and actions. This allows the quantitative application of this goal-free utility to agent control problems with subjective information only. This results in promising self-motivated and self-organized behaviour, as seen in [1, 2, 15, 18, 19, 21].

In this paper we focus entirely on the continuous domain, since it encompasses a lot of important control problems and robotic actuation. Previous work of Jung [15] has shown how empowerment can be computed for the continuous domain via discrete random sampling, but the process is very time consuming, making it impractical for typical applications.

We present an alternative, faster approach, by approximating this class of problems by a linear, continuous channel capacity computation. In particular, this new

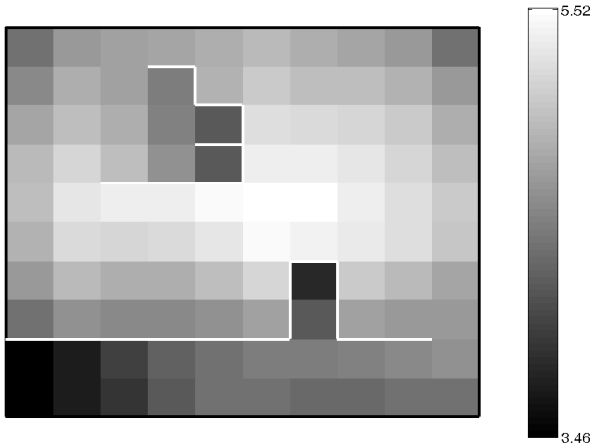

Fig. 1. The graph depicts the empowerment values for 5 step action sequences for the different positions in a 10 × 10 maze. Walls are shown in white, and cells are shaded according to empowerment. As the key suggests empowerment values are in the range [3.46, 5.52] bits. This figure demonstrates that by simply assessing its options (in terms of movement possibilities), the agent can discover implicit features of the world. The most empowered cells in the labyrinth are those that can reliable reach the most positions within the next 5 steps. The graph is a reproduction of the results in [19]

approach allows us to deal with systems with continuous actions, and we can now adjust the horizon of the empowerment computation continuously. As a downside, this approach requires a local linear approximation of the system.

1.1. *Overview*

First we will briefly reflect on the motivation of empowerment as a goal-independent utility function, and its uses for self-organized behaviour generation. We will then formally introduce empowerment as a quantitative measure and the two existing methods of computing it in the discrete and continuous domain respectively.

In the next section we will then demonstrate that empowerment can instead be approximated by computing the capacity for parallel, linear, Gaussian channels if the linear transformation matrix between actuators and sensors is known.

The simulation of the simple pendulum will then be used to demonstrate how this method can be applied to an actual continuous system. For this method to work, one needs to obtain the transformation matrix describing the dynamics of the system. We will show, on the one hand, how to derive it from a given mathematical model and, on the other hand, how one can obtain it empirically, via linear regression on available samples. Furthermore, we will look at the resulting pendulum control, and how the parameters of the empowerment calculation affect the

pendulum behaviour.

In the next section we then compare the resulting empowerment landscapes for the Gaussian channel method to those obtained by Jung's existing method, including a variation that also relies on empirical sampling and binning. Finally, all the methods are discussed regarding their specific benefits and shortcomings in view of the application of empowerment to guide self-organization in a continuous agent-environment system.

2. Related Work

It is now well understood that embodied agents already receive an adaptive and evolutionary advantage by virtue of their embodiment alone [24]. While many forms of adaptation and learning require some external goal-orientated supervision, critique, or perspective, embodiment provides a vehicle for self-determination which does not necessitate such external goals. The agent's perception of the world it is embodied in, with regard to the worlds structure and dynamics, provides significant prior structure to inform the agent and facilitate its decision making processes. The actions resulting from decisions made are only limited by the agents perception, complexity, and degree to which it has adapted to its environment. Much recent work concentrates on modelling information structure in such a perception-action loop [17, 22, 7, 20, 29].

This appreciation of the virtues of embodiment has prompted the development of many techniques to exploit such guided self-organisation of agent behaviour. Homeokinesis [13] is a predictive methodology that combines evolution and learning, and drives an embodied agent to achieve a better understanding of its own embodiment. The aim for the agent is to sustain a smooth controlled behaviour, but unlike homeostasis [14], an older cybernetic perspective on intrinsic motivation, which attempts to preserve a stationary state, homeokinesis aims to produce a stable kinetic regime.

In living organisms decision making is of course the province of neural pathways, and information theory has long provided a mechanism for analysing the efficiency and redundancy inherent in sensory stimuli [4, 6]. Information theory is considered to form the basis of an ecological theory of sensory processing [3], ecological in the sense that the information theory can be used to assess the neural response resulting from the stimulus environment. Hence, it is hypothesised that agents or organisms benefit from optimising informationally the sensory and neural configurations they apply to their environment. Predictive information-based methodologies [25, 8, 5] demonstrate one way in which information theory can provide such intrinsic motivation for an agent. Additionally, the concept of "flow" [12] from psychology has been used to provide a vehicle for intrinsic motivation in machine learning [26, 28] and related fields [16, 30]. In this work, another approach is adopted, namely that of empowerment [18, 19, 21, 1, 2, 15], an information-theoretic utility function which is universal in the sense that it is independent from a specific external task. Em-

powerment derives solely from the perception the agent has of its environment, and the means it has to affect that environment (its actions).

2.1. *Motivation for Empowerment*

It is generally accepted in the game of chess, that moving a knight to the outer squares of the board limits the knight's mobility and its ability to affect the game, a phenomena known colloquially as "a knight on the rim is grim". A knight on the edge of the board has fewer potential moves to make, it has lower empowerment. Empowerment can therefore be seen as a measure of mobility, but may be even further generalised colloquially as the tendency of an agent or organism to keep its options open [21], whether its options concern mobility, food, reproduction, or any other means by which an organism or agent can exert control over its environment.

2.2. *Information Theory Fundamentals*

Empowerment is formalized using terms from information theory, first introduced by Shannon [27]. For self-containedness, we introduce the relevant information-theoretic notions. The first information theoretic quantity to understand is *entropy*, which is a measure of uncertainty. Entropy is defined as

$$H(X) = -\sum_{x\in\mathcal{X}} p(x)\log p(x) \tag{1}$$

where X is a discrete random variable with values $x \in \mathcal{X}$, and $p(x)$ is the probability mass function such that $p(x) = Pr\{X = x\}$. Throughout this paper base 2 logarithms are used by convention, and therefore the resulting units are in *bits*. Introducing another random variable Y, jointly distributed with X, enables the definition of the *conditional entropy*

$$H(X|Y) = -\sum_{x\in\mathcal{X}} p(y)\sum_{y\in\mathcal{Y}} p(x|y)\log p(x|y). \tag{2}$$

This measures the remaining uncertainty about X if Y is known. Since Eq. (1) is the general uncertainty of X, and Eq. (2) the remaining uncertainty if Y has been observed, their difference, called *mutual information*, quantifies the information one can gain about X by observing Y. Mutual information is defined as

$$I(X;Y) = H(Y) - H(Y|X). \tag{3}$$

The mutual information is symmetric (see [11]), since

$$I(X;Y) = H(Y) - H(Y|X) = H(X) - H(X|Y). \tag{4}$$

Considering the classical communication problem of transmitting a signal over a channel, essentially there is a sender and a receiver. The sender transmits a signal, denoted by the random variable X, and the receiver receives a potentially different signal, denoted by the random variable Y. The communication channel defines how

the transmitted signal is transformed into the received signal. In the case of discrete signals, the channel itself is described by the conditional probability distribution $p(y|x)$. Mutual information (Eq. (3), (4)) may then be interpreted as the amount of information, on average, that the received signal contains about the transmitted signal. The *channel capacity* is then defined as the maximum mutual information for the channel over all possible distributions $p(x)$ of the transmitted signal

$$C = \max_{p(x)} I(X;Y). \tag{5}$$

Hence the channel capacity is defined as the maximum amount of mutual information the received signal Y can contain about the transmitted signal X. Mutual information is calculated using $p(x)$ and $p(y|x)$, but channel capacity is calculated on $p(y|x)$ alone, as $p(x)$ is determined by the maximization criterion (Eq. (5)).

2.3. *Empowerment Formalism*

Empowerment is an information theoretic quantity which represents the capacity of the *perception-action loop* [18]. The perception-action loop formalism considers the whole system as consisting of sensor (S_t), actuator (A_t) and rest of system (R_t) at time t. These components and the dependencies between them evolve through time and can be illustrated with a Bayesian network (Fig. 2).

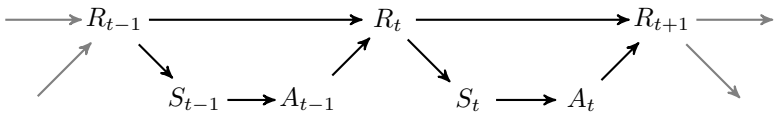

Fig. 2. The perception-action-loop visualised as a Bayesian network. S is the sensor, A is the actuator, and R represents rest of the system.

R_t is included to formally account for the effects of the actuation on the future sensoric input. In terms of the classical communication problem, R_t is the state of the actuation channel.

Empowerment is defined for stochastic dynamical systems where transitions arise as the result of making a decision, e.g. such as an agent interacting in an environment. Here a vector-valued state space $\mathcal{S} \subset \mathbb{R}^D$ and a discrete action space $\mathcal{A} = \{1, ..., N_A\}$ are assumed. The transition function is given by $p(\mathbf{s}_{t+1}|\mathbf{s}_t, a_t)$ and describes the probability of transitioning from state $\mathbf{s}_t$ to state $\mathbf{s}_{t+1}$ when the agent makes action decision a_t. The system is fully defined for such 1-step actions[a],

[a]Note that the actual channel in general depends on the current state of the world, i.e. on R_t, so, strictly spoken, we consider a particular channel at a particular R_t and hence empowerment will depend on R_t. For ease of notation, however, we will not write this dependence on R_t in the following derivation.

and empowerment may be defined as the channel capacity of the agent's actuation channel terminating at the sensor [19]

$$\mathfrak{E}_t := C\left(p(\mathbf{s}_{t+1}|\mathbf{s}_t, a_t)\right) = \max_{p(a_t)} I(\mathcal{S}_{t+1}; \mathcal{A}_t|\mathbf{s}_t). \tag{6}$$

Instead of using a single action as the transmitted signal, we are interested in more general n-step actions. Thus for $n > 1$, we define an n-step action sequence as $\vec{a}_t^n := (a_t, ..., a_{t+n-1})$, and the transition function then becomes $p(\mathbf{s}_{t+n}|\mathbf{s}_t, \vec{a}_t^n)$. Thus n-step empowerment is defined as:

$$\mathfrak{E}_t := C\left(p(\mathbf{s}_{t+n}|\mathbf{s}_t, \vec{a}_t^n)\right) = \max_{p(\vec{a}_t^n)} I(\mathcal{S}_{t+n}; \mathcal{A}_t|\mathbf{s}_t) \tag{7}$$

Empowerment is measured in *bits*. Empowerment has a number of interpretations: one can consider it as the number of distinguishable options available to an agent [18]. An agent attempting to maximize empowerment as it moves, attempting to maximize its available options at any time. Another interpretation is that of an information-theoretic analogue of the concept of combined "controllability/observability" known from control theory. Empowerment measures the amount of (Shannon) information that an agent can potentially "inject" into the environment via its actions and recapture later. It is important to note that it only identifies *potential* information injection, not what the agent actually ends up doing.

In this paper, for simplicity all agents are considered to have perfect knowledge of the environment, although imperfect information and its effects on the empowerment state-space will be discussed later. This implies that R_t and S_t are the same and makes it possible to define empowerment purely in terms of state transitions, i.e. in terms of states $\mathcal{S}$ and their successors $\mathcal{S}'$ and actions $\mathcal{A}$. Hence, using Eq. (3) and (5), the empowerment $C(\mathbf{s})$ of a particular state $\mathbf{s}$ may be defined as the Shannon channel capacity (Eq. (5)) between $\mathcal{A}$, the action selection, and $\mathcal{S}'$, the resulting successor state. By making substitutions for entropy and conditional entropy in terms of actions, states, and successor states into Eq. (5), it can be shown that empowerment can be written as[b]:

$$C(\mathbf{s}) := \max_{p(\vec{a_v})} \sum_{v=1}^{N_n} p(\vec{a}_v) \int_{\mathcal{S}} p(\mathbf{s}'|\mathbf{s}, \vec{a}_v) \log p(\mathbf{s}'|\mathbf{s}, \vec{a}_v) d\mathbf{s}' \tag{8}$$

For further details on the derivation of Eq. (8), refer to [15]. Note that actions have been assumed discrete, hence the sum over the actions, but states are continuous, therefore the density integral over the states. In this way, the perception-action loop formalism is treated as an interpretation of the classical communication problem. With the perception-action loop, a discrete memoryless communication channel,

[b]For notational convenience, instead of writing $p(\mathbf{s}_{t+n}|\mathbf{s}_t, \vec{a}_t^n)$ we will now just write $p(\mathbf{s}'|\mathbf{s}, \vec{a})$ to denote the transition from state $\mathbf{s}$ to state $\mathbf{s}'$ under action sequence $\vec{a}$. We will also use the parameter v to loop over the actions of $\vec{a}$

there exist algorithms to calculate the channel capacity, for example the iterative algorithm Blahut-Arimoto [9] which is discussed in the next section.

2.4. *Blahut-Arimoto Algorithm*

The Blahut-Arimoto algorithm (BA) [9] is an expectation maximization (EM) type algorithm for computing the channel capacity given by Eq. (8). BA iterates over distributions $p_k(\vec{a})$, where k is the iteration parameter, converging towards the distribution that maximises Eq. (8). Since a discrete action space is assumed, $p_k(\vec{a})$ can be represented by a vector $p_k(\vec{a}) \equiv (p_k^1, ..., p_k^{N_n})$. We follow the general notation from [15], and define the variable $d_{v,k}$ as:

$$d_{v,k} := \int_{\mathcal{S}} p(\mathbf{s}'|\mathbf{s}, \vec{a}_v) \log \left[\frac{p(\mathbf{s}'|\mathbf{s}, \vec{a}_v)}{\sum_{i=1}^{N_n} p(\mathbf{s}'|\mathbf{s}, \vec{a}_i) p_k^i} \right] d\mathbf{s}'. \tag{9}$$

BA begins with initialising $p_0(\vec{a})$ to be uniformly distributed, by simply setting $p_0^v = \frac{1}{N_n}$ for all actions $v = 1, ..., N_n$ (action sequences for multiple step empowerment). At each iteration $k \geq 1$, the new approximation for the probability distribution $p_k(\vec{a})$ is obtained from the old one $p_{k-1}(\vec{a})$ using

$$p_k^v := z_k^{-1} p_{k-1}^v \exp(d_{v,k-1}) \tag{10}$$

where z_k^{-1} is a normalisation parameter ensuring that the approximation for the probability distribution $p_k(\vec{a})$ sum to one for all actions $v = 1, ..., N_n$, and is defined as

$$z_k := \sum_{v=1}^{N_n} p_{k-1}^v \exp(d_{v,k-1}). \tag{11}$$

Thus $p_k(\vec{a})$ is calculated for iteration step k, it can be used to obtain an estimate $C_k(\mathbf{s})$ for the empowerment $C(\mathbf{s})$ using

$$C_k(\mathbf{s}) = \sum_{v=1}^{N_n} p_k^v \cdot d_{v,k}. \tag{12}$$

The algorithm can be iterated over a fixed number of times or until the absolute difference $|C_k(\mathbf{s}) - C_{k-1}(\mathbf{s})|$ drops below an arbitrary chosen threshold ϵ.

The remaining issue with using BA for continuous space is the evaluation of the high-dimensional integral in $d_{v,k}$ (Eq. 9). The next section outlines two different Monte-Carlo(MC) based approaches for addressing this issue.

2.5. *Empowerment in Continuous Space*

To calculate empowerment from Eq. (8) in the continuous domain, we can employ different methods. First this section will outline how empowerment may be calculated by discretising the continuous domain using binning. Secondly, an approach that assumes that $p(\mathbf{s}'|\mathbf{s}, \vec{a}_v)$ may be approximated by a multivariate Gaussian distribution is presented.

2.5.1. *Monte-Carlo Binning Approach*

Binning is a useful technique for approximating the continuous state space as it does not rely on making assumptions about the underlying distributions. However, care should be taken with any binning approach to ensure that where possible each bin contains approximately the same number of samples to ensure no bias is inadvertently applied [23]. Binning results in replacing the conditional probability densities $p(\mathbf{s}'|\mathbf{s}, \vec{a})$ by regular probabilities $p(\tilde{\mathbf{s}}'|\mathbf{s}, \vec{a})$. Once the continuous data has been binned, the BA algorithm can be applied to the resulting conditional distribution, substituting the high-dimensional integral in Eq. (9) with a summation over all bins. Resulting empowerment landscapes derived using this method for the simple pendulum will be presented later.

2.5.2. *Monte-Carlo Multivariate Gaussian Approach*

The Monte-Carlo binning approach has several drawbacks, one being that the binning can introduce artefacts stemming from the arbitrary way in which bins are allocated. However, the main drawback is that it requires many bins to be used to get an accurate representation of empowerment. This requirement of many bins places a significant additional computational load on an already computationally costly methodology. For this reason, in [15] a Monte-Carlo Multivariate Gaussian Approach was used. To introduce it, we essentially follow the exposition from [15].

In this approach, the assumption that $p(\mathbf{s}'|\mathbf{s}, \vec{a}_v)$ is a multivariate Gaussian, or can be reasonably well-approximated by it, is made, i.e.

$$\mathbf{s}'|\mathbf{s}, \vec{a_v} \sim \mathcal{N}(\mu_v, \mathbf{\Sigma}_v) \tag{13}$$

where $\mu_v = (\mu_{v,1}, ..., \mu_{v,D})^T$ is the mean of the Gaussian and the covariance matrix is given by $\mathbf{\Sigma}_v = diag(\sigma^2_{v,1}, ..., \sigma^2_{v,D})$. The mean and covariance will depend upon the action $\vec{a}_v$ and the state $\mathbf{s}$. Samples from the distribution will be denoted $\tilde{\mathbf{s}}$ and can be generated using standard algorithms.

The following algorithm summarises how to compute the empowerment $C(\mathbf{s})$ given a state $\mathbf{s} \in \mathcal{S}$ and transition model $p(\mathbf{s}'|\mathbf{s}, \vec{a}_v)$:

(1) **Input:**
   (a) Specify state $\mathbf{s}$ whose empowerment is to be calculated.
   (b) For every action $v = 1, ..., N_n$, define a (Gaussian) state transition model $p(\mathbf{s}'|\mathbf{s}, \vec{a}_v)$, which is fully specified by its mean μ_v and covariance Σ_v.

(2) **Initialise:**
   (a) $p_0(\vec{a}_v) := 1/N_n$ for $v = 1, ..., N_n$.
   (b) Draw N_{MC} samples $\tilde{\mathbf{s}}'_{v,i}$ each, according to distribution density $p(\mathbf{s}'|\mathbf{s}, \vec{a}_v) = \mathcal{N}(\mu_v, \mathbf{\Sigma}_v)$ for $v = 1, ..., N_n$.
   (c) Evaluate $p(\tilde{\mathbf{s}}'_{v,i}|\mathbf{s}, \vec{a}_\mu)$ for all $v = 1, ..., N_n$; $\mu = 1, ..., N_n$; and sample $i = 1, ..., N_{MC}$.

(3) **Iterate** the following variables for $k = 1, 2, ...$ until $|c_k - c_{k-1}| < \epsilon$ or the maximum number of iterations is reached:

(a) $z_k := 0$, $c_{k-1} := 0$

(b) For $v = 1, ..., N_n$

i. $d_{v,k-1} := \frac{1}{N_{MC}} \sum_{i=1}^{N_{MC}} \log \left[\frac{p(\tilde{\mathbf{s}}'_{v,i}|\mathbf{s}, \vec{a}_v)}{\sum_{j=1}^{N_n} p(\tilde{\mathbf{s}}'_{v,i}|\mathbf{s}, \vec{a}_j) p_{k-1}(\vec{a}_j)} \right]$

ii. $c_{k-1} := c_{k-1} + p_{k-1}(\vec{a}_v) \cdot d_{v,k-1}$

iii. $p_k := p_{k-1}(\vec{a}_v) \cdot \exp(d_{v,k-1})$

iv. $z_k := z_k + p_k(\vec{a}_v)$

(c) For $v = 1, ..., N_n$

i. $p_k(\vec{a}_v) := p_k(\vec{a}_v) \cdot z_k^{-1}$

(4) **Output:**

(a) Empowerment $C(\mathbf{s}) \approx c_{k-1}$ (estimated).

(b) Distribution $p(\vec{a})$ achieving the maximum mutual information.

As with the MC binning approach, results for the MC multivariate Gaussian approach for the simple pendulum will be presented later.

3. Empowerment Approximation in Continuous State Space

In this section we introduce a faster method to compute empowerment for a continuous, but locally linear domain. We will show how the more general problem of computing channel capacity in the continuous domain, given some specific assumptions, can essentially be reduced to parallel Gaussian channels, where channel capacity can be determined with well-established algorithms.

3.1. *Continuous, Locally Linear Empowerment*

Let S be a multi-dimensional, continuous random variable defined over the vector space $\mathbb{R}^n$. Let A be a multidimensional random variable defined over $\mathbb{R}^m$. We will call A the action variable, and S the perception variable, and we assume that there is a linear transformation $T : \mathbb{R}^m \to \mathbb{R}^n$ that defines the relation of those variables as

$$S = TA + Z. \tag{14}$$

Z is another multi-dimensional, random variable defined over $\mathbb{R}^n$, modelling the noise in the system. Z is independent of A and S. Each dimension $q \leq n$ of Z is independent of each other dimension, and has a normal distribution with $Z_q \sim \mathcal{N}(0, N_q)$ for each dimension. A possible explanation for this noise, if we are dealing with an agent, would be the measurement inaccuracy introduced by the agent's sensors.

What we want to calculate again is the channel capacity

$$C = \max_{p(a):E(A^2)<P} I(S;A). \tag{15}$$

The power constraint P is introduced to limit the values A can assume, otherwise the channel capacity could be made arbitrarily large, by allowing sufficiently large action amplitudes to render all outcomes distinguishable. The power constraint can model a "physical" power constraint as a conceptual limitation of action amplitudes (i.e. deviations from the "neutral" action). Generally, we will not assume a necessarily physical interpretation of power, but rather a conceptual one.

3.2. *MIMO channel capacity*

If we assume, in addition to our assumption of independent noise, that the variance of the noise in each dimension is 1, then the problem is similar to computing the channel capacity for a linear, multiple input, multiple output channel (MIMO) with additive Gaussian noise.

This can be solved by standard methods [31], namely by applying a Singular Value Decomposition (SVD) to the transformation matrix T, that decomposes T as

$$T = U\Sigma V^* \tag{16}$$

where U and V are unitary matrices and Σ is a diagonal matrix with non-negative real values on the diagonal. This allow us to transform Eq. (14) to

$$U^*S = \Sigma V^* A + U^* Z. \tag{17}$$

Each dimension of the resulting variables U^*S, ΣV^*A and U^*Z can be treated as an independent channel (see [31]), reducing this to computing the channel capacity for linear, parallel channels with added Gaussian noise, as in [11],

$$C = \max_{P_i} \sum_i \frac{1}{2} \log\left(1 + \frac{\sigma_i P_i}{E\left[(U^*Z)_i^2\right]}\right) = \max_{P_i} \sum_i \frac{1}{2} \log(1 + \sigma_i P_i) \tag{18}$$

where σ_i are the singular values of Σ, and P_i is average power used in the i-th channel, following the constraint that

$$\sum_i P_i \leq P. \tag{19}$$

Since the channel capacity achieving distribution is a Gaussian distribution, this means the optimal input distribution is a Gaussian with a variance of P_i for each channel. We can simplify Eq. (18) because the expected value for the noise is 1.0, since the unitary matrix applied to Z does not scale, but only rotates the input, so it retains its original variance of 1.0.

The optimal power distribution that maximizes Eq. (18) can then be found with the water-filling algorithm [11].

3.3. *Transformation of Noise*

If we assume that the noise $Z_i \sim \mathcal{N}(0, N_i)$ is Gaussian and independent in each dimension, but has different variances N_q for each channel, we cannot easily remove the noise from Eq. (18) after the transformation with U^*. Rotating the noise would introduce covariances between the different noise distributions for each channel.

If we want to make sure that the noise distributions are still independent after being transformed we could ensure that they are spherical (having the same variance in each dimension) before they are transformed. Assuming independent, but non-spherical noise distributions

$$Z_i \sim \mathcal{N}(0, N_i) \tag{20}$$

we now define a diagonal matrix D as

$$D = \begin{pmatrix} d_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & d_n \end{pmatrix} \text{ with } d_i = \frac{1}{\sqrt{N_i}}. \tag{21}$$

If all the values for N_i are positive, non-zero values, then D is a non-singular diagonal matrix, with positive, non-zero diagonal values. Scaling a continuous random variable with a scalar s changes the information contained in that variable to $H(sX) = H(X) + \log(s)$. Mutual Information remains unaffected, so if we multiply S, the random variable that results from our actions with the scaling matrix D, it would do nothing to S's informational content about A. Thus it follows that

$$I(S; A) = I(DS; A) = I(DTA + DZ; A). \tag{22}$$

By replacing T, the transformation from A to S with DT we create a channel capacity problem with the same channel capacity, but with spherical, independent noise. It can then be solved with the standard algorithm outlined in the last section which relies on independent noise.

The contribution of the different noise levels to the channel capacity are not lost but merely included in the matrix DT. Realizing this also makes it easier to compute the original solution, because we do not need to keep track of the different noise levels in $E\left[(U^*Z)_i^2\right]$, since the resulting channel capacity is now only dependent on the singular values of DT.

3.3.1. *Noise with Zero Variance*

Some discussion concerning the treatment of noise is in place. We remind the reader that in a non-degenerate deterministic continuous scenario, i.e. a scenario without noise, different action sequences will in general lead to different states, thus empowerment will be maximal and equal to $\log|A|$ (with $|A|$ the number of action sequences).

Since in the present Gaussian model the action space is continuous, there are infinitely many action sequences. The ensuing empowerment value will thus be infinite, unless the noiseless degrees of freedom are not affected by the actions.

Only the presence of noise induces an "overlap" of outcome states that allows one to obtain meaningful empowerment values. However, this is not a significant limitation in practice, as virtually all applications need to take into account actuator, system and/or sensor noise.

We now generally assume that parallel noise on the output channel is transformed away by the procedure from Sec. 3.3. Therefore, in our examples, unless otherwise noted we assume that the variance of noise is one, and the mean of the noise zero. This is without loss of generality; any non-zero mean could be immediately transformed away since any affine transformation in the system would leave the mutual information unaffected.

4. Experiments (Pendulum)

In this section we will discuss a simple but illustrative experiment: the simple pendulum balancing task. The pendulum task will showcase how our new approximation procedure can be applied to non-linear models via linear approximation. We will demonstrate two different methods to obtain the linear transformation matrix.

We will then compare the results of our approximation with Jung's more generic empowerment estimation (see Sec. 2.5.2), as well as with the modified version that relies on binning to obtain the resulting action distributions (see Sec. 2.5.1).

4.1. *Matrix Calculation*

In order to approximate the empowerment landscape for the state space of the simple pendulum we calculate the matrix that describes how the system transforms control inputs into successor states for all possible starting points in the state space. The rest of the section gives the slightly tedious calculation of the relevant matrices, to make clear how the formalism from Sec. 3.2 is to be applied here. The reader less interested in technical details can skip directly to the result just before Sec. 4.1.1.

Consider a pendulum, its current state at the time t defined by its angle: ϕ, and its angular velocity $\dot{\phi}$

$$s_t = \begin{pmatrix} \phi_t \\ \dot{\phi}_t \end{pmatrix}. \tag{23}$$

We now want to look at the development of the pendulum state for three time steps[c], all of which are of duration Δt. We approximate behaviour of the pendulum by assuming that it develops linearly within the duration of each time step.

[c]The choice of three time steps stems from the fact that this is the smallest number of steps which will give non-trivial results. Thus the computation must necessarily extend beyond a simple linear approximation of the successive step.

The agent can apply an action in form of a control input u_t which is added to the acceleration of the pendulum during each time step. The next state of the pendulum at $t + \Delta t$ can then be computed as

$$s_{t+\Delta t} = s_t + A_t \Delta t + B u_t \Delta t \tag{24}$$

with the matrices A_t and B as

$$A_t = \begin{pmatrix} \dot{\phi}_t \\ \frac{g}{l} \sin \phi_t \end{pmatrix}, \tag{25}$$

$$B = \begin{pmatrix} 0 \\ 1 \end{pmatrix}. \tag{26}$$

For the next step we need to calculate a new A-matrix, $A_{t+\Delta t}$, since the nonlinearity of the system causes A to depend on the new s. The values can be computed as:

$$A_{t+\Delta t} = \begin{pmatrix} \dot{\phi}_{t+\Delta t} \\ \frac{g}{l} \sin \phi_{t+\Delta t} \end{pmatrix} \tag{27}$$

A similar matrix $A_{t+2\Delta t}$ can be computed for the third step. Straightforward insertion and reformulation allow us to compute the actual values for $s_{t+3\Delta t}$ as

$$\phi_{t+3\Delta t} = \Delta t \Phi_2 + \Phi_1 + 3\Delta t u_t + 2\Delta t u_{t+\Delta t}, \tag{28}$$

$$\dot{\phi}_{t+3\Delta t} = \Phi_2 + \frac{g\Delta t}{l} \sin(\Phi_1 + 2\Delta t u_t) + \Delta t u_t + \Delta t u_{t+\Delta t} + \Delta t u_{t+2\Delta t}. \tag{29}$$

with Φ_1 and Φ_2 representing the following terms that do not depend on any of the control inputs u:

$$\Phi_1 = \Delta t \left(\frac{g\Delta t}{l} \sin(\phi_t) + 2\dot{\phi}_t \right) + \phi_t, \tag{30}$$

$$\Phi_2 = \frac{g\Delta t}{l} \sin(\Delta t \dot{\phi}_t + \phi_t) + \frac{g\Delta t}{l} \sin \phi_t + \dot{\phi}_t. \tag{31}$$

We now want to express the values of $x_{t+3\Delta t}$ as a linear equation of the following form, where K is a constant matrix, whose value only depends on the starting state s_t:

$$s_{t+3\Delta t} = K + T \begin{pmatrix} u_t \\ u_{t+\Delta t} \\ u_{t+2\Delta t} \end{pmatrix}. \tag{32}$$

Since $\phi_{t+3\Delta t}$ is already in linear form, there is nothing left to do here. The only problem is the sine term for $\dot{\phi}_{t+3\Delta t}$ in Eq. (29). Therefore, we use the Euler form of the Taylor approximation to linearise part of Eq. (29). $\dot{\phi}_{t+3\Delta t}$ can be expressed as a sum of functions, where all functions but $f_1(u_t)$ are linear:

$$\dot{\phi}_{t+3\Delta t} = f_1(u_t) + f_2(u_{t+\Delta t}) + f_3(u_{t+2\Delta t}) + \Phi_2. \tag{33}$$

We approximate f_1 around $u_t = 0$ as

$$f_1(u_t) \approx f_1(0) + \frac{d}{du_t} f_1(0)(u_t - 0), \tag{34}$$

$$f_1(u_t) \approx \frac{g\Delta t}{l} \sin(\Phi_1) + u_t \Delta t \left(\frac{(\Delta t)^2 g}{l} \cos(\Phi_1) + 1 \right). \tag{35}$$

Since f_1 is now linearised, we can now write $s_{t+3\Delta t}$ as

$$\begin{pmatrix} \phi_{t+3\Delta t} \\ \dot{\phi}_{t+3\Delta t} \end{pmatrix} = \begin{pmatrix} \Delta t \Phi_2 + \Phi_1 \\ \frac{\Delta t g}{l} \sin(\Phi_1) + \Phi_2 \end{pmatrix} + \Delta t \begin{pmatrix} 2\Delta t & \Delta t & 0 \\ \frac{(\Delta t)^2 g}{l} \cos(\Phi_1) + 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} u_t \\ u_{t+\Delta t} \\ u_{t+2\Delta t} \end{pmatrix}. \tag{36}$$

The singular values of the matrix T can then be used to calculate the channel capacity for the pendulum:

$$T = \begin{pmatrix} 2\Delta t & \Delta t & 0 \\ \frac{(\Delta t)^2 g}{l} \cos(\Delta t(\frac{g\Delta t}{l} \sin(\phi_t) + 2\dot{\phi}_t) + \phi_t) + 1 & 1 & 1 \end{pmatrix}. \tag{37}$$

The reason we have to cover at least three time steps is that fewer steps would fail to capture the non-linearity of the system since it does not allow change in the first input to propagate through all variables. The resulting approximated empowerment landscape would essentially be constant[d].

More steps are possible, but their calculation is omitted for reasons of brevity. The matrix T for 4 steps is

$$T = \begin{pmatrix} \frac{\Delta t^3 g}{l} \cos\Phi_1 + 3\Delta t & 2\Delta t & \Delta t & 0 \\ \frac{2\Delta t^2 g}{l} \cos(\Phi_2 \Delta t + \Phi_1) + \frac{\Delta t^2 g}{l} \cos\Phi_1 + 1 & \frac{\Delta t^2 g}{l} cos(\Phi_2 \Delta t + \Phi_1) + 1 & 1 & 1 \end{pmatrix}. \tag{38}$$

Note, all resulting matrices only have two rows, and thereby a maximum of two singular values. The power distribution that maximises channel capacity can therefore be calculated analytically for this example.

4.1.1. *Resulting Control*

To control the actuation of the pendulum we implement a 1-step greedy algorithm, which chooses an action in the present state that will maximize the empowerment of the resulting state.

The simulation used to test this algorithm computes the evolution of the pendulum in time steps of 100 Hz. At each time step the current velocity is dampened

[d]A more detailed and systematic characterization of the interaction between system nonlinearity and empowerment landscape is envisaged, but is outside of the scope of the present paper and will be undertaken in future work.

by the factor 0.00005 (which is not reflected in the empowerment model), and all the simulation we discuss here start with the pendulum in the lower rest position.

At each time step the algorithm knows which state s_t the pendulum is in, and decides how much power is applied to the pendulum actuator over the next time step. For every step this can be up to the maximum power of $\sqrt{P_i}$, measured in meters per second square. We remind the reader again that this is not "power" in the physical sense and is therefore not measured in units of work per time. The overall power constraint is given as P, the sum of the P_i for the respective time steps i.e. three or four. Since we are working with the four-step matrix, there are four power values for the four successive actions.

First, we will define possible n actuator input candidates as the evenly spaced values between $\sqrt{P_i}$ and $-\sqrt{P_i}$ (n is chosen as an odd number, to allow zero actuation as an action). Negative $-\sqrt{P_i}$ is a full powered actuator input into the opposite angle direction than P_i. Note that the choice of n actions is a parameter of the greedy control algorithm that uses empowerment as a utility function, it does not affect the calculation of empowerment itself, which will still be performed based on the assumption of continuous actions.

For each of those values we then compute how the system would develop if this input were applied continuously for the next 50 time steps [e]. The empowerment of each resulting state (after 50 time steps) is then computed by inserting the state's parameters into the 4-step matrix from Eq. (38). The actuator input for the next single time step is the one which leads to the highest empowered state after 50 time steps. After one time step this calculation is performed again, this time extrapolating from the current state.

The resulting control shows how the pendulum traverses the state space (Fig. 3). In this case, the pendulum accelerates with full power in one direction, until it hits the point were its acceleration is not powerful enough to carry it any higher. Then it accelerates with full power into the opposite direction, swinging through the rest point, up to the highest possible point on the other side. This is repeated until it reaches the top, were it decelerates before reaching the apex, so it comes to rest right in the topmost position.

For an underpowered pendulum, this is the optimal strategy for reaching the top position. Should the pendulum be so underpowered that it would not be able to reach the top, due to dampening for example, it would then end up in the highest possible periodic oscillation.

Note that this results purely from greedily optimizing the resulting empowerment value of the control action. There is no hard-coded incentive or reward for the

[e]We choose to make a greedy selection of actions based on the states that would be reached within 50 time steps (or 0.5 seconds) to capture the non-linear nature of the system. In essence, we wanted the simulation to be more fine-grained than the control algorithm. The main difference this longer lock-ahead introduces is a smoothness in the resulting trajectories. A one step greedy algorithm still produces similar pendulum upswing, but there are sharper turns in the trajectories

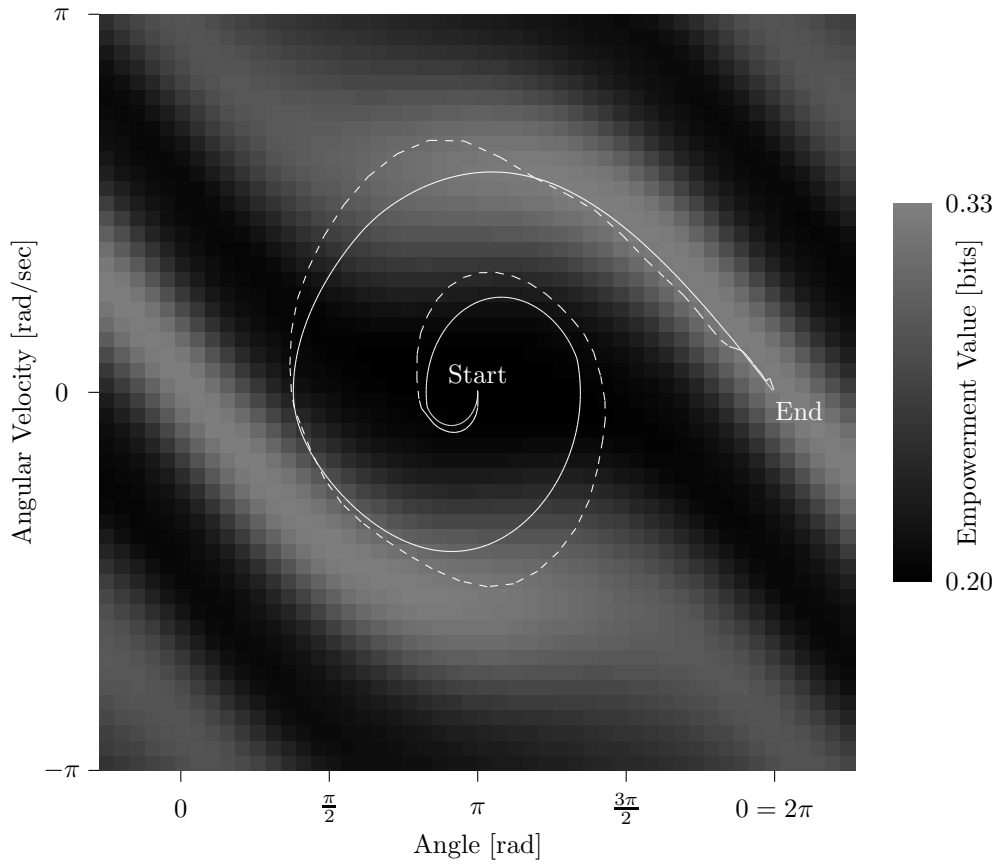

Fig. 3. Graph depicting the state space of a pendulum and its associated empowerment values. The solid line shows the trajectory of a pendulum controlled by a greedy empowerment maximization algorithm based on the underlying Gaussian quasilinear empowerment landscape. The dashed line shows the controlled pendulum trajectory based on a greedy maximisation of the Monte Carlo Gaussian approach.

pendulum to reach the topmost rest position. Rather, the top position appears to be an advantageous state to start in if required to reach a larger number of states reliably in the imminent time horizon.

Additionally, note that the greedy algorithm does not implement a gradient ascent in the empowerment landscape, because the dynamics of the system impose certain state changes. If the pendulum has a high velocity in a given direction, then there will be a large change in pendulum angle in the successor state regardless of what the control input is. Hence the pendulum control is not necessarily able to climb up on a ridge in the empowerment landscape, if the dynamics are moving the state away from that ridge of high empowerment.

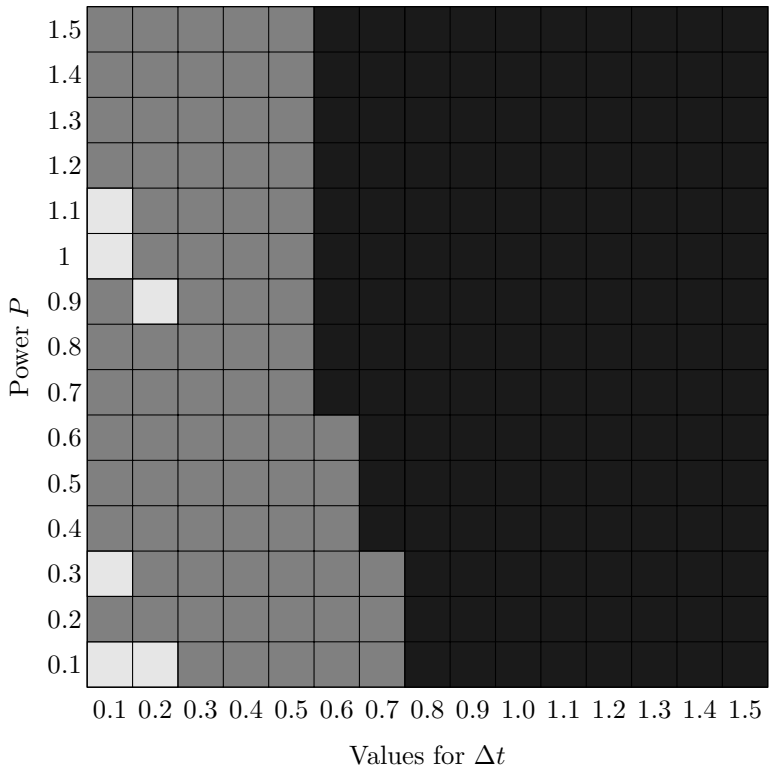

Fig. 4. A plot of the pendulums behaviour for different values of Δt and P, with the pendulum starting in the lower rest position. The white area results in oscillation, the grey indicates reaching the upper rest position, and in the black area the pendulum remains in the lower position.

4.2. *Variable Parameters*

Once the dynamics of the system are defined, the empowerment landscape produced by our new approximation algorithm depends essentially only on two parameters, the time step length Δt, and the power constraint P. If we vary through those parameters, as seen in Fig. 4, we can observe three different classes of behaviour:

- The pendulum swings up and comes to a controlled rest in the upper position.
- The pendulum swings up continuous to oscillates
- The pendulum remains in the lower rest position

Note that only the power constraint is an actual parameter of the pendulum system, while Δt is only relevant for the control algorithm. So all the entries in Fig. 4 for similar power deal with a similar system. Since every power has at least one entry that manages to reach the upper rest position, it is possible for all examined power levels to get there. So the differences in the behaviour of systems with similar power constraints result only from the different empowerment landscapes that form the

basis of our control algorithm, and not from any differences in the actual pendulum system.

4.2.1. *Variation of the Power Constraint*

A closer look at the different underlying empowerment landscapes in Fig. 5 shows their changes in regard to power constraint P and time step length Δt.

In general, an increase in power will result in an increase in empowerment, no matter where in the state space the system. This is not immediately visible, since the colouring of the graphs is normalized, so the black and white correspond to the lowest and highest empowerment value in that sub graph, respectively.

A more interesting effect is a potential *inversion* of the empowerment landscape. Inversion means that for two specific points in the state space it might be that for one power level the first has a higher empowerment than the other, but for a different power level this relationship is reversed, and now the second has a higher empowerment.

This is a result of how the capacity is distributed on the separate parallel channels. Be reminded, each channel i contributes its own amount to the overall capacity

$$C = \max_{P_i} \sum_i \frac{1}{2} \log(1 + \sigma_i P_i) \tag{39}$$

subject to a total power constraint P. Depending on the different values for σ_i, power is first allocated to the channel with the highest amplification value σ_i, up to a point were the return in capacity for the invested power diminishes so much that adding power to a different channel yields more capacity. From that point on the overall system acts as if it was one channel of bigger capacity.

So, for low power, the factor that determines the channel capacity is the value of the largest σ alone. Once the power increases, the values of both the σ become important. It is therefore possible that for low power, a point with one large σ has comparatively high empowerment, while for a higher power level, another point has a higher empowerment, because the combination of all the σ is better. This case is what actually happens in the pendulum example. In Fig. 5 we can consider the row of landscapes for a Δt of 0.7. With increasing power there appears a new ridge of local maximal empowerment around the lower rest position of the pendulum. This actually causes the pendulum to remain in the lower rest position for the examples with higher power.

It is striking that this effect somewhat coincides with the transition from an underpowered pendulum to one that can easily reach any point in its state space without any upswing manoeuvres. If this proves indeed to be the case, it would imply that this change in the computed empowerment landscape actually reflects a true change in fundamental qualitative characteristics of the model. We suspect that this observation may offer a key for a more thorough interpretation of the phenomenon in the future.

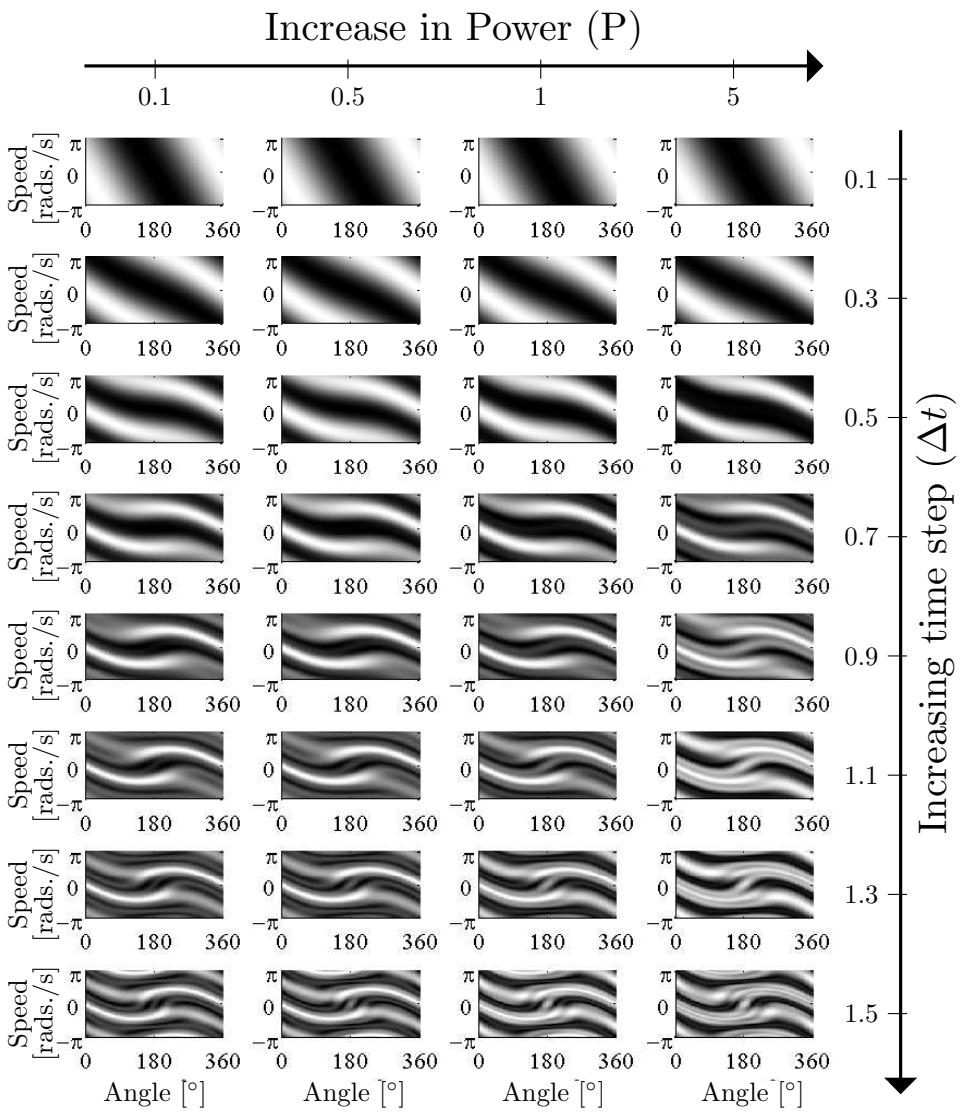

Fig. 5. A visualization of the different empowerment landscapes resulting from computation with different parameters for time step length Δt and power constraint P. All computations are based on the analytically derived four-step transformation matrix T from Eq. (38).

4.2.2. *Variation of the Time Step Duration*

Another parameter we can vary when computing the empowerment landscape is Δt, the time step length. This parameter does not change the dynamics of the simulation

itself, but it is a variable that characterizes the interaction between controller and system, and thus has considerable influence on the computation of empowerment itself.

Since we are dealing with a stepwise linear approximation, an overly long time step size will make the approximation worse. In our pendulum simulation this results in some high frequency patterns (such as $\Delta t = 1.5$, in Fig. 5) which also seem to be, according to Fig. 4, damaging to the control algorithm.

On the other hand, as the value for Δt becomes infinitesimal small, several of the terms in our matrix from Eq. (38) vanish. The resulting empowerment landscape approximates $\sin(\phi_t)$. While this might be a better approximation for a very short look into the future, and it also retains the upper rest position as a point of high empowerment, it turns out not to be very helpful for our greedy control algorithm. In the worst case the pendulum is actuated towards the top, up to that point were gravity compensates for the force of actuation. The actuator still tries to move the pendulum higher, and so the pendulum remains in that point of equilibrium, unable to find a path through the state space that will end up in the upper rest position.

For larger time step duration the pendulum displays the behaviour described earlier, which can also be seen in Fig. 3. The pendulum swings up and reverses the actuation at the apex, gathering even more energy. Eventually it reaches the high rest position. Approaching the rest position the control algorithm then breaks the pendulum to not overshoot the upper position. This is helped by the fact that for larger Δt the ridges of high empowerment leading towards the upper rest position define a good approach to the top rest position. If the time step duration decreases it becomes harder for the pendulum to approach and decelerate, since the guiding "ridges" of high empowerment vanish (See Fig. 5). Lacking the predictive power of larger time steps, successful deceleration now depends on how closely the trajectory of the pendulum leads to the upper rest position. Sometimes, when Δt is too small, the pendulum just overshoots and enters some oscillating behaviour, were it swings around fully (See Fig. 3). This happens at different levels of power, because varying the power influences how the pendulum initially approaches the top.

Basically, Δt defines the horizon of our empowerment calculation. The smaller it gets the worse it becomes at predicting the future. A good value for Δt should therefore balance the need for prediction with the need for good approximation.

4.3. *Approximation via Sampling*

In general it will not always be possible to rely on a fully known mathematical model to obtain the transformation matrix T analytically. An alternative, which is compatible with the Gaussian channel approximation, would be the reconstruction of an approximated linear transformation matrix empirically via linear regression.

Consider a system available as a black box simulation, or a real world experiment that can be repeated several times. Where this is not possible, one can often still operate under the "ergodic" assumption, meaning one replaces repeated separate

runs starting in a state by the statistics obtained from a single run that keeps revisiting the same state sufficiently often for the statistics. This assumption even includes an agent who finds itself in a similar (according to its sensors) position, and has a record of different actions it took, and the respective outcomes.

In any case, we assume now that m samples of transitions have been obtained for a given starting state, consisting of the inputs (actions) and the respective outputs (resulting state). In the pendulum case this would be the applied power u for the length of the four time steps each, and the two values (speed $\dot{\phi}$ and angle ϕ) of the resulting pendulum state.

If we put all the inputs into a $m \times 4$ matrix F (the four actions from one sample form a row), and all the outputs into a $m \times 2$ matrix G (each row is a sample of the resulting angle and velocity), we can then compute a linear transformation matrix $\hat{T}$ which minimizes the least squares error as:

$$\hat{T} = (F^T F)^{-1}(F^T G) \tag{40}$$

We performed the linear approximation for two different sampling methods, random sampling and regular sampling. For regular sampling there were five different actuation levels for each time step (full actuation each direction, half actuation in each, no actuation), creating $5^4 = 625$ samples. For the random sampling we also created 625 samples, but for each of their 4 time steps their actuation was chosen from a continuous, uniform, random distribution bound by the maximal actuations in each direction. The resulting mappings $\hat{T}$ after linear regression were used to create the empowerment landscapes in Fig. 6.

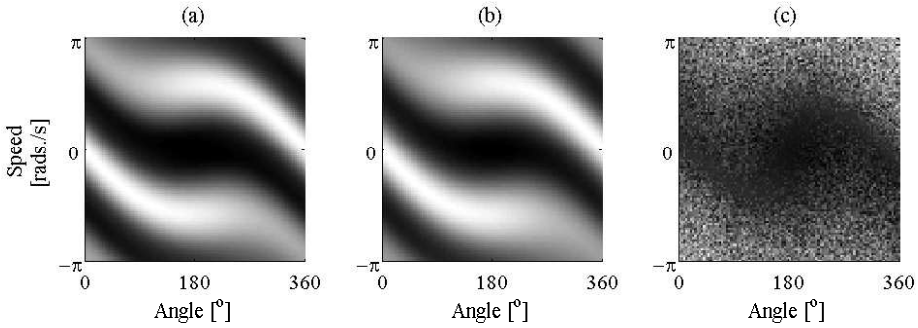

Fig. 6. Three graphs depicting the different empowerment landscapes obtained with a. analytical computation, b. regular sampling and c. random sampling. All simulation were for 4 time steps with 0.7 second, with $P = 0.1$.

We see that the regular sampling closely resembles the landscape created with the analytical model. The random sampling, however, contains a lot of noise for the same number of samples, even so it still retains a very rough qualitative similarity in the empowerment landscape; especially the low empowered zone around the

lower rest position. A pendulum actuated with this landscape will actually manage to leave the lower rest position, but fails in most cases to stabilize in the upper position. It is, however, clear that, whenever there is a choice, regular sampling is preferable.

4.4. *Comparison of Empowerment Landscapes*

This section evaluates the different methodologies for calculating empowerment for the pendulum. In general the discernment of empowerment landscapes can be affected greatly by the limitations of sensors as shown by Fig. (7). Although in this paper sensor accuracy is assumed to be perfect (the global state of the world is known to the agent), a degradation of the empowerment landscape can be observed with some instances of the Monte-Carlo binning methodology. In particular, Fig. 7 shows the same 1-step empowerment landscape all generated using 36 angle bins and 0, 2, and 20 speed bins respectively. It can be seen from Fig. 7 (a) that ignoring the angular speed state results in a simplistic representation of the empowerment landscape. Introducing 2 speed bins, one for positive speed and one for negative, begins to transform the empowerment landscape as shown by Fig. 7 (b). Fig. 7 (c) shows the effects of incorporating 20 speed bins with the angle bins. Fig. 7 (d) shows the 1-step empowerment landscape generated using the MC multivariate Gaussian approach [15]. Comparing Fig. 7 (c) and (d), one can see that the empowerment landscapes are similar, increasing the number of speed bins in Fig. 7 plot (c) will only make them more so, but the resulting computational cost with the binning approach is even more severe than with the MC multivariate Gaussian approach.

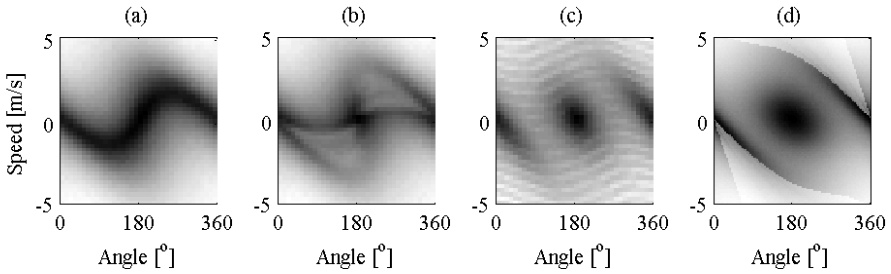

Fig. 7. Comparison of the two MC methodologies, demonstrating that incorporating increased knowledge of the state space improves the discernment of the empowerment landscape; (a) shows 1-step empowerment using the binning approach when only angle states are binned and angular speed states are ignored; (b) shows that when two angular speed state bins are introduced along with the angle state bins that the empowerment landscape becomes more than a simple sinusoid; (c) demonstrates that the empowerment landscape that is obtained the more speed state bins are added, approximates the Multivariate Gaussian approach in plot (d). Suggesting that the Multivariate Gaussian approach provides the most accurate representation of the empowerment landscape for the pendulum, with the binning methodology only providing comparable results for large numbers of bins.

Figure 8 shows a three dimensional representation of the 3-step empowerment landscape generated using the MC multivariate Gaussian approach. The figure shows the same landscape with various orientations from birdseye (top left) and then from left to right progressively more towards cross-section viewpoints. The figure highlights the extent of the nonlinearity of the empowerment landscape. As can be seen from the cross-sectional plots, the gradient change in empowerment values are steep. The resulting landscape is filled with some local minima, but contains one global minimum corresponding to the angle that the pendulum is in the downward resting position ($\theta = 180^o$).

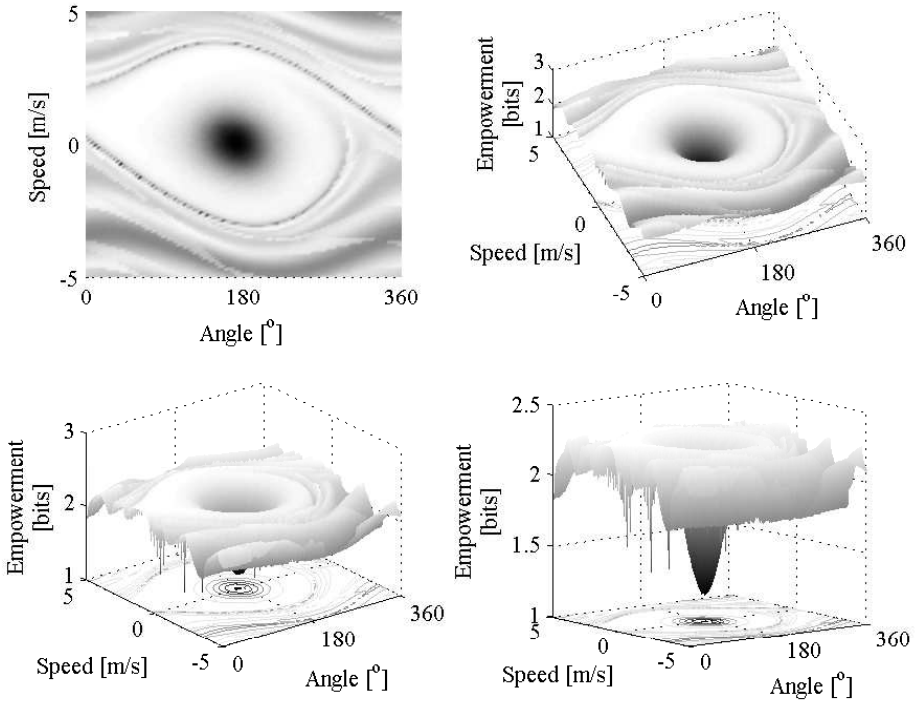

Fig. 8. 3 dimensional representation of Monte Carlo multivariate Gaussian evaluation of the 3-step empowerment landscape for the simple pendulum.

Figure 9 shows a comparison between the MC multivariate Gaussian approach and the Gaussian channel approximation. On the top row of the figure (plots (a) to (c)) the MC multivariate Gaussian approach was used to generate the empowerment landscapes for a fixed time step of 0.2 seconds for 1- to 3-step empowerment respectively. On the bottom row of Figure 9 (Plots (d) to (f)) the Gaussian channel approximation was employed for powers 0.1, 0.5, 0.9, and Δt 0.5, 0.9 and 1.3 respec-

tively. Both methodologies calculate the empowerment landscape for future action selection, they both employ different interpretations of the empowerment *horizon*. In general it is difficult to compare the two methodologies in the sense that the first method uses a finite time step and hence has a fixed empowerment calculation horizon in time. The Gaussian channel approximation, on the other hand, uses a combination of power (P) and time step (Δt) to determine the empowerment horizon. Nevertheless the two methodologies can be compared qualitatively with one another for particular choice of parameters. In either case, if one greedily maximises the action selection based on relative empowerment values when travelling through the pendulum state-space, the resultant control is very similar, as the majority of the peaks and troughs of the empowerment landscapes for the two methodologies coincide.

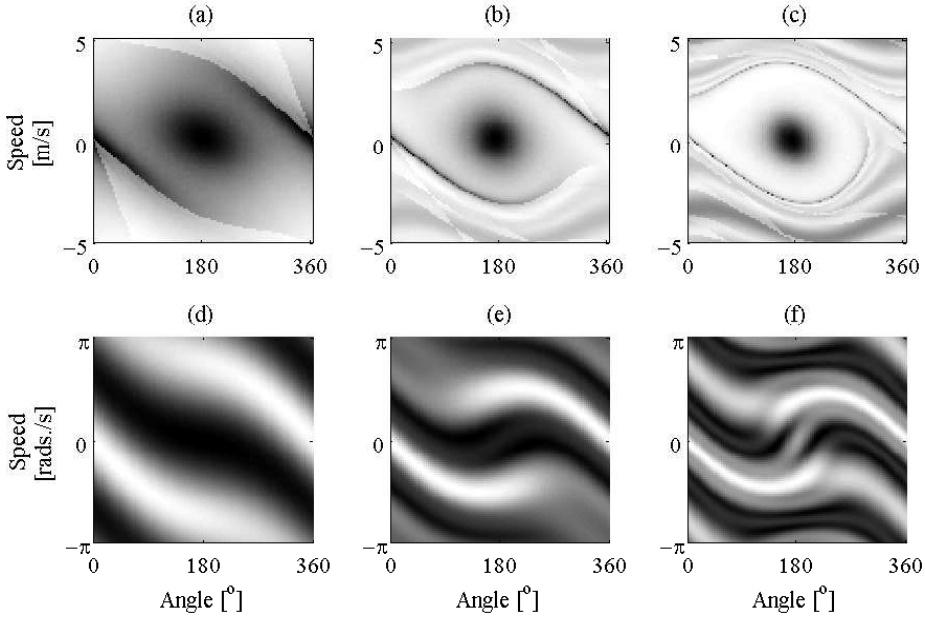

Fig. 9. Comparison between MC multivariate Gaussian approach (a,b,c) and the Gaussian channel approximation (d,e,f). Plots (a) to (c) generated using a fixed time step of 0.2 seconds for 1 to 3-step empowerment. Plots (d) to (f) generated for powers 0.1, 0.5, 0.9, and Δt 0.5, 0.9 and 1.3 respectively

5. Discussion

Comparing the different methods to approximate empowerment it seems that while they differ in their detailed empowerment landscapes, they nonetheless create similar overall structures and behaviour. Looking at all four approximation methods,

in particular, for the low-powered pendulum we see that all of them assign:

- a low empowerment value to the lower rest position
- a high empowerment value to the upper rest position
- high values to a path leading towards the upper rest position

The last one seems to be particularly important for the resulting behaviour, since it allows our greedy control approaches to find a trajectory through the state space that leads towards the upper rest position. A naive approach, where the acceleration just tries to directly move the pendulum towards the top all the time simply does not work in the underpowered case.

Most problems, both for the pendulum behaviour and for the similarity of the approximation arise when either power or the time step length are getting too large. This is probably caused by two different factors.

Firstly, since we are only dealing with approximations, it seems likely that large time steps make the linear approximations more susceptible to error, and likewise, high power levels can also create larger errors in the approximations.

Secondly, since empowerment is defined in regard to the agent's sensors, several parameters not only change the numerical approximation, by which empowerment is calculated, but actually change the value *per se*. For example, if we are binning the values as the actual sensors quantize the values, then a calculation based on binning might actually be closer to the actual empowerment value than assuming that the agent senses the resulting states continuously.

Similar considerations have to be taken for agent's with vastly different power levels. An agent with high power might actually have a different empowerment landscape than one that only has little actuation power. The same can be said for the temporal horizon the agent is considering. There might be something just behind the temporal horizon that completely changes the landscape (such as the death of the agent), and therefore different horizons can actually have radically different empowerment landscapes. So, basically, the approximation of the original problem becomes worse, not because of some systematic error, but because the system is actually determining the empowerment for a different system.

When comparing the control strategies resulting from the quasilinear approach and the Monte Carlo Gaussian approach , there is a paradigm difference between the power and time step parameters with the quasilinear approach, and the magnitude of the control input with the Monte Carlo approach. This can be clearly seen from Fig. 3, which shows that although the phase and angular velocity are similar, there are some minor differences because of this. Additionally, there are some further minor differences that result from different underlying empowerment maps.

A remark is in place concerning the generality of the results. To introduce the Gaussian quasilinear approximation to empowerment, we have exclusively concentrated on the pendulum scenario and compared it to earlier methods in this particular scenario.

How it will fare in other, more general, scenarios needs to be investigated further, but it should be mentioned that the older methods have been shown to successfully operate in a variety of more complex and involved scenarios [1, 2, 10, 15, 18, 19, 21]; this gives rise to hopes that the quasilinear Gaussian approximation will be able to inherit the advantageous qualities of the other approximation methods, even when transferred to other scenarios.

The most critical problem for the quasilinear approximation (see also Future Work) is the case of hard transitions of dynamics (such as collisions). The more general (but slower) empowerment approximation methods give appealing and intuitive results in these cases [1, 2, 18, 19, 21], but how the quasilinear approximation will fare here and whether it needs to be modified to keep pace with the other methods, remains to be seen.

5.1. *Performance Comparison*

As discussed in Sec. 5, the presented methods are not exactly computing the same underlying empowerment, and variable parameters in each method cannot be equated directly. Therefore, it is not easy to create a meaningful quantitative comparison in terms of a percentile increase in running time. But the qualitative differences in performance are so large that this is not necessary.

Computing the full empowerment landscape for more than two time steps in the future with the Blahut/Arimoto-based MC binning, or MC Multi-variant Gaussian approach takes days. Computing the same landscape based on the SVD of the linear transformation matrix happens in less than one second. If the matrix is derived beforehand with the mathematical model, the greedy control algorithm runs in real time. If the matrix T has to be computed via linear regression, it is slightly slower, mostly due to the amount of simulated sampling being done.

Another comparative advantage of both the MC binning, and the linear regression based matrix decomposition, is its wider applicability, since they can extract models from sampled data, which might be obtained via repeated experimentation, or from some black box simulation model.

6. Future Work

The work presented demonstrates that fast empowerment calculation that can be used in a real-time system (such as actual robotic control) is in the realm of the possible. This is in contrast to the traditional methods which, while demonstrating desirable properties of empowerment, were too slow for practical application. To proceed further in this direction two main challenges have to be addressed: the representation of real world energy constraints, and the ability to deal with discontinuities in the system.

Model extraction also becomes relevant, since more complex control systems do not necessarily offer enough structural insight for an observer to construct explicit equations describing them. Therefore it becomes necessary to either construct the

transition matrix T for every point in the state space from empirical data, or to construct a model for the overall system from data, and automatically derive the transition matrix for each point.

Especially hard discontinuities in the dynamics, such as collisions, can make it arbitrarily difficult to model the relationship between input and output as a linear transformation, making the approximation error uncontrollably large. A decomposition into separately treated linear approximations might be able to circumvent this.

At the same time, the underlying empowerment concept is completely unaffected by discreteness or continuity of the system it deals with. It would therefore be desirable to to seamlessly include discrete system dynamics that is completely non-linear in nature.

Finally, we conclude that it would be desirable to transfer some of the generality of the underlying empowerment concept into methods for empowerment calculation in systems combining continuous and discrete dynamics, especially systems of real-world relevance.

Acknowledgments

This research was supported by the European Commission as part of the CORBYS (Cognitive Control Framework for Robotic Systems) project under contract FP7 ICT-270219. The views expressed in this paper are those of the authors, and not necessarily those of the consortium.

7. References

References

[1] Anthony, T., Polani, D., and Nehaniv, C., On preferred states of agents: how global structure is reflected in local structure, *Proc. Artificial Life XI. MIT Press* (2008) 25–32.

[2] Anthony, T., Polani, D., and Nehaniv, C., Impoverished empowerment: meaningful action sequence generation through bandwidth limitation, *Advances in Artificial Life. Darwin Meets von Neumann* (2011) 294–301.

[3] Atick, J., Could information theory provide an ecological theory of sensory processing?, *Network: Computation in neural systems* **3** (1992) 213–251.

[4] Attneave, F., Some informational aspects of visual perception., *Psychological review* **61** (1954) 183.

[5] Ay, N., Bertschinger, N., Der, R., Güttler, F., and Olbrich, E., Predictive information and explorative behavior of autonomous robots, *The European Physical Journal B-Condensed Matter and Complex Systems* **63** (2008) 329–339.

[6] Barlow, H., Sensory mechanisms, the reduction of redundancy, and intelligence, *The mechanisation of thought processes* (1959) 535–539.

[7] Bertschinger, N., Olbrich, E., Ay, N., and Jost, J., Autonomy: An information theoretic perspective, *Biosystems* **91** (2008) 331–345.

[8] Bialek, W., Nemenman, I., and Tishby, N., Predictability, complexity, and learning, *Neural Computation* **13** (2001) 2409–2463.

[9] Blahut, R., Computation of channel capacity and rate-distortion functions, *Information Theory, IEEE Transactions on* **18** (1972) 460–473.
[10] Capdepuy, P., Polani, D., and Nehaniv, C., Maximization of potential information flow as a universal utility for collective behaviour, in *Artificial Life, 2007. ALIFE'07. IEEE Symposium on* (IEEE, 2007), pp. 207–213.
[11] Cover, T. M. and Thomas, J. A., *Elements of Information Theory*, 99th edn. (Wiley-Interscience, 1991).
[12] Csikszentmihalyi, M., *Beyond boredom and anxiety.* (Jossey-Bass, 2000).
[13] Der, R., Steinmetz, U., and Pasemann, F., *Homeokinesis: A new principle to back up evolution with learning* (Max-Planck-Inst. für Mathematik in den Naturwiss., 1999).
[14] Harvey, I., Homeostasis and rein control: From daisyworld to active perception, in *Proceedings of the Ninth International Conference on the Simulation and Synthesis of Living Systems, ALIFE*, Vol. 9 (2004), pp. 309–314.
[15] Jung, T., Polani, D., and Stone, P., Empowerment for continuous agent environment systems, *Adaptive Behavior* **19** (2011) 16.
[16] Kaplan, F. and Oudeyer, P., Maximizing learning progress: an internal reward system for development, *Embodied artificial intelligence* (2004) 629–629.
[17] Klyubin, A., Polani, D., and Nehaniv, C., Organization of the information flow in the perception-action loop of evolved agents, in *Evolvable Hardware, 2004. Proceedings. 2004 NASA/DoD Conference on* (IEEE, 2004), pp. 177–180.
[18] Klyubin, A., Polani, D., and Nehaniv, C., All else being equal be empowered, *Advances in Artificial Life* (2005) 744–753.
[19] Klyubin, A., Polani, D., and Nehaniv, C., Empowerment: A universal agent-centric measure of control, in *Evolutionary Computation, 2005. The 2005 IEEE Congress on*, Vol. 1 (IEEE, 2005), pp. 128–135.
[20] Klyubin, A., Polani, D., and Nehaniv, C., Representations of space and time in the maximization of information flow in the perception-action loop, *Neural Computation* **19** (2007) 2387–2432.
[21] Klyubin, A., Polani, D., and Nehaniv, C., Keep your options open: an information-based driving principle for sensorimotor systems, *PloS ONE* **3** (2008) e4018.
[22] Lungarella, M., Pegors, T., Bulwinkle, D., and Sporns, O., Methods for quantifying the informational structure of sensory and motor data, *Neuroinformatics* **3** (2005) 243–262.
[23] Olsson, L., Nehaniv, C., and Polani, D., Sensor adaptation and development in robots by entropy maximization of sensory data, in *Computational Intelligence in Robotics and Automation, 2005. CIRA 2005. Proceedings. 2005 IEEE International Symposium on* (IEEE, 2005), pp. 587–592.
[24] Pfeifer, R., Bongard, J., and Grand, S., *How the body shapes the way we think: a new view of intelligence* (The MIT Press, 2007).
[25] Prokopenko, M., Gerasimov, V., and Tanev, I., Evolving spatiotemporal coordination in a modular robotic system, *From Animals to Animats 9* (2006) 558–569.
[26] Schmidhuber, J., Exploring the predictable, *Advances in evolutionary computing* **6** (2002) 579–612.
[27] Shannon, C. E., A mathematical theory of communication, *Bell Sys. Tech. Journal* **27** (1948) 623–656.
[28] Singh, S., Lewis, R., Barto, A., and Sorg, J., Intrinsically motivated reinforcement learning: An evolutionary perspective, *Autonomous Mental Development, IEEE Transactions on* **2** (2010) 70–82.
[29] Sporns, O. and Lungarella, M., Evolving coordinated behavior by maximizing information structure, in *Artificial life X: proceedings of the tenth international conference*

on the simulation and synthesis of living systems (2006), pp. 323–329.
[30] Steels, L., The autotelic principle, *Embodied Artificial Intelligence* (2004) 629–629.
[31] Telatar, E., Capacity of multi-antenna gaussian channels, *European transactions on telecommunications* **10** (1999) 585–595.