Max-Planck-Institut für Mathematik in den Naturwissenschaften Leipzig

Apportionment of work among environment, body and brain of an agent

(revised version: February 2021)

by

Carlotta Langer and Nihat Ay

Preprint no.: 2 2021

Apportionment of work among environment, body and brain of an agent

Carlotta Langer[1] and Nihat Ay[1,2,3]

[1] Max Planck Institute for Mathematics in the Sciences, Leipzig, Germany

[2] Leipzig University, Leipzig, Germany

[3] Santa Fe Institute, Santa Fe, USA

An embodied agent, that performs a goal-directed action, influences its environment and gets influenced by it. Hence information flows not only among the agents control mechanisms, but also between the agent and its environment. In this article we combine different methods in order to create a framework in which we are able to closely examine the information flow among the body, brain and environment of an agent. We test this framework in a simple experimental setup. There the agents do not learn to perform a task, but we calculate the optimal behavior using the method Planning as Inference, in which the information geometric em-algorithm is used to optimize the likelihood of the goal. Then we analyze the resulting distribution using information theoretic methods including measures corresponding to the concepts of Morphological Computation and Integrated Information Theory. Comparing the behavior of these measures under changing morphological circumstances highlights the asymmetric relationship between these two concepts.

keywords: Information Theory, Information Geometry, Planning as Inference, Morphological Computation, Integrated Information, Embodied Artificial Intelligence

1 Introduction

1.1 Objective

An agent that is faced with a task can solve it using solely its brain, its bodies interaction with the world or a combination of those. This article presents a framework to analyze the importance of these different interactions for an embodied agent. To illustrate the idea we discuss the following scenario:

Consider a sailor at sea without any navigational equipment. The sailor has to rely on the information given by the sun or the visible stars in order to determine in which direction to steer. The more complex part of the task is solved by the information processed in the brain of the sailor. On the other hand, a bird equipped with magneto-reception, meaning one that is able to use the magnetic field of the earth to perceive its direction, can rely on this sense and does not need to integrate a lot of different sources of information. Here the body of the bird interacts with the environment for the bird to orientate itself. The complexity of the task is met by the morphology of the bird. Taking this example further we are able to consider a modern boat with a highly developed navigation system. The sailor now only needs to know how to interpret the machines and will therefore have less complex calculations to do. The complexity of the task shifts from the brain and background knowledge of the sailor towards the construction of the navigation system, which receives and integrates different information sources for the sailor to use. Therefore the environment changes in a way that makes the task easier.

Our objective is to analyze this shift of complexity. We will do that by quantifying the importance of the information flow in an embodied agent performing a task under different morphological circumstances. The setting of our experiment will be presented in Section 2. The question we ask is: How is the complexity of solving the task distributed among the different parts of the body, brain and environment?

The main statements that we will support by our experiments are:

1. The importance of the information flow in the controller of an embodied agent depends additionally on the information flowing to and from the controller.

2. The more the agent can rely on the interaction of its body with the environment to solve a task, the less important the information flow in the controller becomes. Equivalently, if most of the task can be solved by the controller, then the body and environment become less important.

In order to test these statements, we need to develop a theoretical background.

1.2 Theoretical Background

We will model the different interactions using the sensori-motor loop, which depicts the different connections between the world W and the controller C, the sensors S and actuators A of an agent. This will be discussed further in Section 2.2.

Using the sensori-motor loop we are able to define a set of probability distributions reflecting the structure of the information flow of an agent interacting with the world. Now we need to find the probability distributions that describe a behavior that optimizes the likelihood of success. It would be possible to use a learning or evolutionary algorithm on the agents to find this optimal behavior, but instead we will apply a method called "Planning as Inference".

Planning as Inference is a technique proposed by Attias in [6], further developed in [33], [32], [34], in which a goal directed planning task under uncertainty is solved by probabilistic inference tools. This method models the actions an agent can perform as latent variables. These variables are then optimized with respect to a goal variable using the em-algorithm, an information geometric algorithm that is guaranteed to converge. This course of action has the advantage that we can directly calculate the optimal policies without having to first train the agents. We will describe this method in the context of our experimental setup in further detail in Section 2.3.

Having calculated the distributions that describe the optimal behavior, we are able to apply diverse information theoretic measures to quantify the strength of the different connections. The measures we are going to discuss are defined by minimizing the KL-divergence between the original distribution and the set of split distributions. The split distributions do not have the information flow that we want to measure. A short introduction to information theoretic concepts like the KL-divergence is given in Appendix 6.1. Following this concept we are able to quantify the strength of the different information flows, which leads to measures that can be seen in the context of Integrated Information, Reactive Control, Morphological Computation and Predictive Information. These measures are defined in Section 3.

Using information theoretic measures to quantify the information flow in an embodied agent is a natural concept, since we could perceive the different parts of the system as communicating with each other. Surely the world does not actively send information to the controller, but the controller still receives information about the world through the sensors. Therefore there have been different studies analyzing acting agents by using information theoretic measures. In [20] for example Klyubin et al. propose maximizing the information flow through the whole system as a learning objective. Furthermore, in [31] Touchette and Lloyd use the concepts of information and entropy to define conditions under which a system is perfectly controllable or observable. Emphazising the importance of the sensory input, Sporns and Pegors utilize entropy and mutual information in [26] to analyze how an agent actively structures its sensory input. Moreover, in [23] Lungarella et al. also include the structure of the motor data in their analysis. The last two cited articles additionally discuss two measures regarding the amount of information and the complexity of its integration. These measures were defined in the context of Integrated Information Theory.

Integrated Information Theory (IIT) proposed by Tononi aims at measuring the amount and quality of consciousness. This theory went through multiple phases of development starting as a measure for brain complexity [30] and then evolved through different iterations [29], [28], towards a broad theory of consciousness [25]. The two key concepts, that stayed consistent, determining consciousness are "Information" and "Integration". Information refers to the number of states a system can be in and Integration describes the amount to which the information is integrated among the different parts of it. Measures for integrated information differ depending on the iteration of the theory they are referring to and on the framework they are defined in. We discussed a branch of these measures building on Information Geometry in [21]. In this article we will use the measure Φ_T for integrated information, since we argue in [21] that this is the true measure in the case of a known environment.

Another consistency in IIT is that these measures focus solely on the brain, meaning on the controller in the case of an artificial agent. Therefore we want to embed these measures into the sensori-motor loop and analyze their behavior in relation to the dynamics of the body and environment. Although the measures are only defined for the controller, there have been tests with evolving embodied agents in the context of Integrated Information Theory. In [15] Edlund et al. measure the values for Integrated Information for simulated evolving artificial agents in a maze and conclude that integrated information grows with the fitness of the agents. Increasing the complexity of the environment leads Albantakis et al. in [1] to the conclusion that Integrated Information needs to grow in order to capture a more complex environment. Albantakis and Tononi go in [2] one step further and conclude from experiments with elementary cellular automata and adaptive logic-gate networks that a high integrated information value increases the likelihood of a rich dynamical behavior. All of these examples focus on the measures in the controller in order to analyze what kind of cause-effect structure makes a difference intrinsically. Since we are interested in an embodied agent solving a task, we want to emphasize the importance of the interaction of the agents body with the world. This leads us to the concept of Morphological Computation.

Morphological Computation describes the reduction of computational complexity for the controller resulting from the interaction between the body and the world [16]. An example for Morphological Computation is the bird using its magneto-reception mentioned earlier in the introduction. Another case of Morphological Computation would be a human grabbing a fragile object compared to a robotic metal hand. The soft tissue of the human hands allows us to be less precise in the calculation of the pressure that we apply. The robot needs to perform more difficult computations and will therefore most likely have a higher integrated information. Does this mean that our experience of this task is less consciousness than the robots? Here we want to take a step back from the abstract concept of consciousness and instead examine the complexity of the tasks. The soft skin of the human hand interacts with the object in a more complicated manner than the robots hand. In this article we want to analyze how the complexity of solving a task is met by the different information flows among the brain, body and environment. Lungarella and Sporns detect in [24] that the information flow in the agent can be affected by changes in the bodies morphology. Examining this phenomenon further we will observe shifts in the importance of the information flows depending on the morphology of the body, which directly changes the complexity of the environment for the agent. We will analyze the changes in Morphological Computation using three different information theoretic measures defined in [17] and [18].

Additionally, we are able to quantify the predictability of the world. We are going to change the length of the agents sensors and therefore directly affect how much of the world the agent perceives and therefore how predictable it is. Following [7] we will measure this as mutual information between the past and the present sensor state. The results of our experiments are presented in Section 4.

2 Setting

In order to analyze the information flow of an acting agent, we examine the following small example. Consider a racetrack as shown in Figure 1. The agents have to move at all times and die as soon as their bodies touch a wall. The design and implementation of the racetrack was done by Nathaniel Virgo. The agents body consists of a blue circle and a blue line marking the back of the agent. The two black lines are binary sensors that only detect whether they touch a wall or not, without reporting the exact distance to the wall. If a sensor touches a wall it turns green and if the body of the agent touches a wall it turns red as we can see below. Although we depicted more than one agent in the environment, these agents do not influence each other.

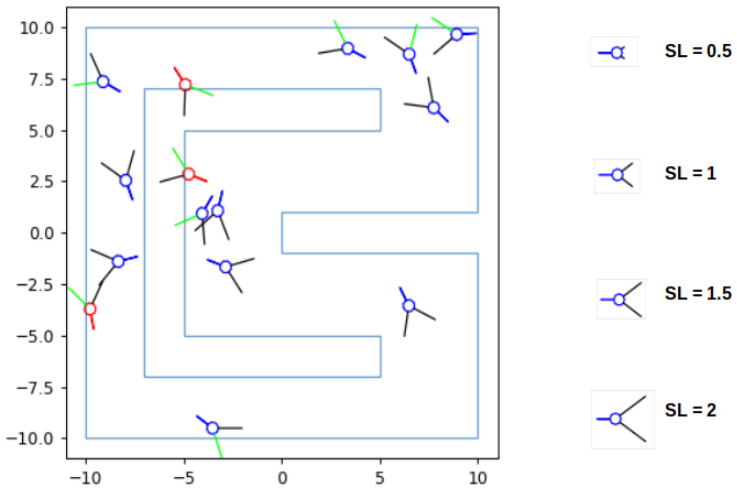


Figure 1: The racetrack the agents have to survive in and the different sensor length, named SL, on the right.

We will vary the length of the sensors of the agents from 0.5 to 2 as shown on the right in Figure 1. The task for the agents is to be alive after their next two movements. The optimal strategy an agent should use for that will be calculated by using the concept of Planning as Inference as discussed in Section 2.3. Using this method we are able to directly determine the optimal behavior without having to train any agents.

Before we discuss this further, we will first present the control architecture of the agents in the next section.

2.1 The Agents

The architecture of the agents is the following one. The agents have two sensors S_t^1, S_t^2, two controllers C_t^1, C_t^2 and two actuator nodes A_t^1, A_2^2. The sensors and controllers send their information to the actuators and controllers in the next point in time. The sensors are only influenced by the world W and the world is only affected by the actuators and the last world state as depicted in 2.

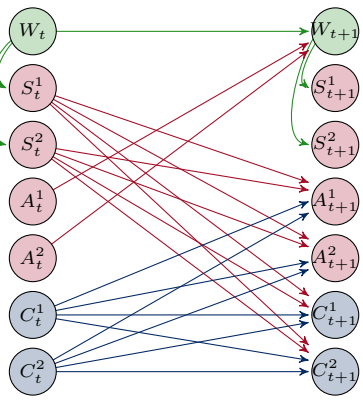


Figure 2: The architecture of the agents.

To simplify we will only draw one node for each S, A and C in the following graphs.

The behavior of the agents is governed by a probabilistic law, which can be model as the following discrete stationary multivariate Markov-Process

$$(X_t)_{t\in\mathbb{N}} = (W_t, S_t, A_t, C_t)_{t\in\mathbb{N}}$$

with the state space $\mathcal{X} = \mathcal{W} \times \mathcal{S} \times \mathcal{A} \times \mathcal{C}$ and the distribution

$$P(x_0, \dots, x_{t+1}) = P(x_0) \prod_{i=1}^{t+1} P(x_t|x_{t-1})$$

$$P(x_{t+1}|x_t) = P(w_{t+1}|w_t, a_t) \prod_k P(s_{t+1}^k|w_{t+1}) \prod_i P(a_{t+1}^i|s_t, c_t) \prod_j P(c_{t+1}^j|s_t, c_t).$$

The corresponding directed acyclic graph is depicted in Figure 3. See [22] for more information on the relationship between graphs and graphical models. Throughout this article we will assume that the distributions on $\mathcal{X}$ are strictly positive.

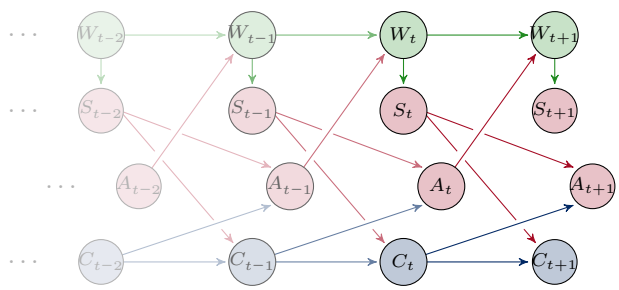


Figure 3: Graphical representation of the Markov-Process $(X_t)_{t\in\mathcal{N}}$.

In the next section we will take a closer look at the role of the environment.

2.2 The Environment

Since we are not able to directly observe the world W, we need to approximate

$$P(w_{t+1}|w_t, a_t) \prod_k P(s_{t+1}^k|w_{t+1})$$

using information intrinsically known to the agent. In order to do that, we will look closer at one step in time $P(x_t, x_{t+1}) = P(x_t)P(x_{t+1}|x_t)$. Here we will make the decision to assume that the environment only

influences the sensors, even in the graph marginalized to one timestep as depicted in Figure 4 on the left. We will then sum over $w_t, w_{t+1} \in \mathcal{W}$ in order to get a Markov Process that only depends on the variables known to the agent.

Proposition 1. *Marginalizing the distribution corresponding to the graph on the left in Figure 4*

$$P(x_t, x_{t+1}) = P(w_t) \cdot P(s_t, a_t, c_t | w_t) \cdot P(w_{t+1} | w_t, a_t) \prod_k P(s^k_{t+1} | w_{t+1}) \prod_i P(a^i_{t+1} | s_t, c_t) \prod_j P(c^j_{t+1} | s_t, c_t)$$

over $(w_t, w_{t+1}) \in \mathcal{W} \times \mathcal{W}$ *leads to the following Markov-Process*

$$P(s_t, a_t, c_t, s_{t+1}, a_{t+1}, c_{t+1}) = P(s_t, a_t, c_t) \cdot \prod_i P(a^i_{t+1} | s_t, c_t) \prod_j P(c^j_{t+1} | s_t, c_t) \cdot \tilde{P}(s_{t+1} | s_t, a_t).$$

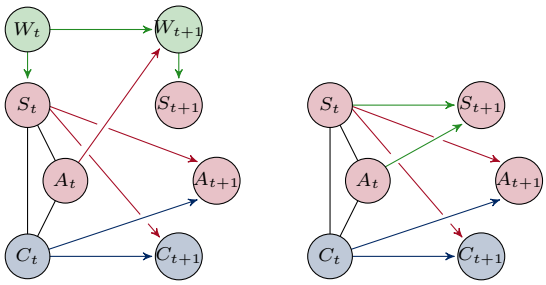


Figure 4: Graphical representation of one timestep and the marginalized graph.

The new process with the approximation of the behavior of the environment is shown in Figure 4 on the right. A similar approximation of the world is also used in [17] in Section 3.3.1. and in [16]. There this approximation is derived by taking $P(S_{t+1}|S_t)$ as the intrinsically available information of $P(W_{t+1}|W_t)$.

Therefore we are able to sample this distribution $\tilde{P}(S_{t+1}, S_t, A_t)$ for every sensor length, meaning we saved 40.000.000 sensor and motor values for agents starting in a random place in the arena. The agents use a simple Hebbian update rule, so that we are able to sample the sensory input of agents moving around. The resulting empirical distributions are shown in Figure 5.

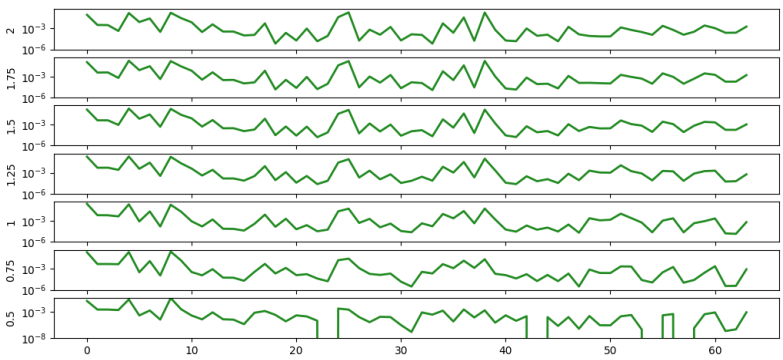


Figure 5: Sampled empirical distribution of the sensor dependencies $\tilde{P}(S_{t+1}, S_t, A_t)$.

In Figure 5 we are able to see, that the length of the sensors has a visible impact on the structure of the sensor distribution. We will use this distribution in order to calculate measures analyzing the interaction between the agent and the environment.

Since we are now able to define a set of distributions that describe the interaction between the agent and the world according to the sensori-motor loop, we will present the method to find the optimal behavior in the next section.

2.3 Optimizing the Behavior

In order to calculate the optimal behavior of the agents, we will use the concept of Planning as Inference. This was originally proposed by Attias in [6] and further developed by Toussaint and collegues in [33], [32] and [34] as a theory of planning under uncertainty. There the conditional distribution describing the action of the agent is considered to be a hidden variable that has to be optimized. This is done by using the EM-algorithm, which is equivalent to the information theoretic em-algorithm in this case. We will describe the em-algorithm below, because of its intuitive geometric nature. The em-algorithm is well known and was proposed by Csiszár and Tusnády in 1984 in [14], further discussed in [5] and [4]. The resulting distribution maximizes the likelihood of achieving the predefined goal.

The goal of the agents in our example is to maximize the probability of being alive after the next two movements. To make at least two steps is necessary since we want that the connection between C_t and C_{t+1} has an impact on the outcome. This can be seen in Figure 7.

We will denote the goal variable by G with the state space $\mathcal{G} = \{0, 1\}$, where $P(g = 1)$ refers to the probability of the agent to be alive. Since the agent moves twice, this distribution depends on the states of the last three sensor and motor commands

$$\tilde{P}(G|S_{t+2}, S_{t+1}, S_t, A_{t+2}, A_{t+1}, A_t).$$

The nodes G depends on are marked with a golden circle in Figure 7.

We sampled this distribution for every sensor length similar to $\tilde{P}(S_{t+1}, S_t, A_t)$ and the resulting empirical distribution is shown in Figure 6. The first 4096 states correspond to situations in which the agent died. Therefore this hints towards the assumption that surviving in this environment becomes easier with increasing sensor length. We will support this observation with the results of the experiments shown in Section 4.

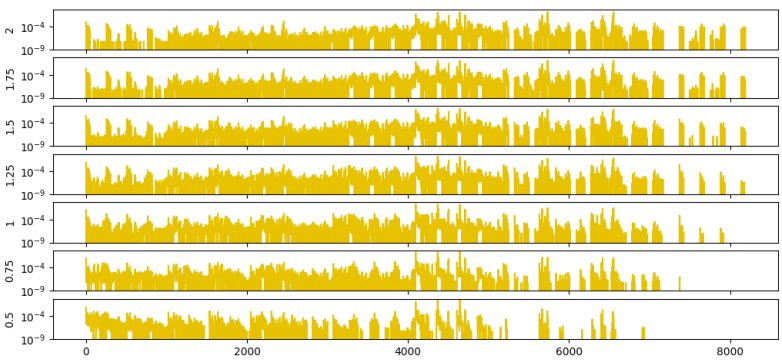


Figure 6: Sampled empirical distribution of the goal variable $P(G, A_t, S_t, A_{t+1}, S_{t+1}, A_{t+2}, S_{t+2})$.

The architecture of the agents considered in this article was discussed in the last sections. There we described how we sample the distribution $\gamma = \tilde{P}(S_{t+1}|S_t, A_t)$ that describes the influence that the agent has on itself through the world. The distributions influencing the actuators and therefore the behavior of the agents are

$$\beta = P(A_{t+1}|S_t, C_t) \qquad \text{and} \qquad \alpha = P(C_{t+1}|S_t, C_t).$$

Hence we will treat (A_{t+1}, C_{t+1}) as hidden variables and optimize their distributions with respect to the goal. We denote these distributions by α, β and γ in order to emphasize the stationarity of the process, meaning that $P(A_{t+1}|S_t, C_t) = P(A_{t+2}|S_{t+1}, C_{t+1})$, $P(S_{t+1}|S_t, A_t) = P(S_{t+2}|S_{t+1}, A_{t+1})$ and $P(C_{t+1}|S_t, C_t) = P(C_{t+2}|S_{t+1}, C_{t+1})$ as indicated in Figure 7.

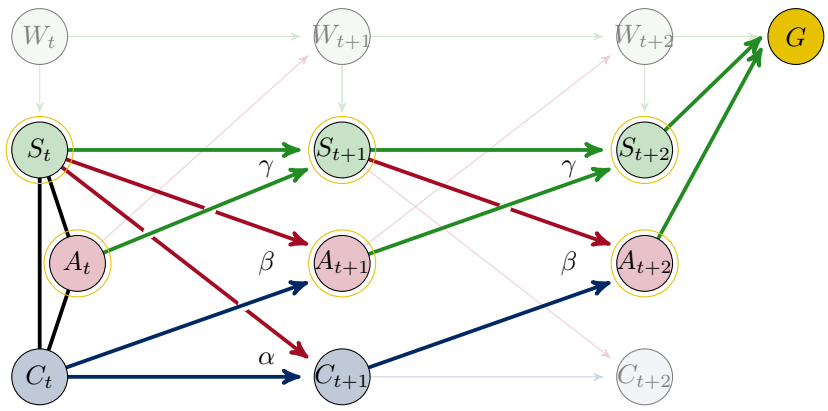


Figure 7: Graphical representation of two timesteps.

It remains to define the initial distribution $\bar{P}(S_t, C_t, A_t)$. We will choose a distribution $\hat{P}(S_t, C_t, A_t)$ at random and then adjust the marginals on (S_t, A_t) such that they are equal to the marginalized sampled distribution $\tilde{P}(S_t, A_t) = \bar{P}(S_t, A_t)$. This can be done by calculating

$$\bar{P}(s_t, a_t, c_t) = \hat{P}(s_t, a_t, c_t)\frac{\tilde{P}(s_t, a_t)}{\hat{P}(s_t, a_t)}.$$

Note that this is an information projection of $\hat{P}$ to the set of distributions defined by having the same (S_t, A_t)-marginals as $\tilde{P}(S_{t+1}, S_t, A_t)$.

Now we are going to define the optimization using the em-algorithm. This algorithm iterates between two sets of distributions in order to find the minimal difference between them. In order to simplify the notation we will define $\mathcal{Y} = \mathcal{S}\times\mathcal{C}\times\mathcal{A}$ and $\mathcal{Z} = \mathcal{Y}\times\mathcal{Y}\times\mathcal{Y}\times\mathcal{G}$, such that $z = (s_t, c_t, a_t, s_{t+1}, c_{t+1}, a_{t+1}, s_{t+2}, c_{t+2}, a_{t+2}) \in \mathcal{Z}$. Let $\mathcal{P}(\mathcal{Z})$ be the set of probability distributions with the state space $\mathcal{Z}$ and let $\mathcal{P}^\circ(\mathcal{Z})$ consist of all the strictly positive distributions in $\mathcal{P}(\mathcal{Z})$. The first set we are considering is

$$\mathcal{M}_G := \{Q \in \mathcal{P}(\mathcal{Z}) | Q(g = 1) = 1,\, Q(g = 0) = 0\}.$$

Every distribution in $\mathcal{M}_G$ achieves the goal state. This goal manifold is a linear family.

The second set consists of all the distributions that factor according to the architecture of the agents, meaning that each of these distributions describes a possible behavior of an agent.

$$\mathcal{M}_A := \Bigg\{ P \in \mathcal{P}^\circ(\mathcal{Z}) | P(z) = \bar{P}(s_t, c_t, a_t)\tilde{P}(s_{t+1}|s_t, a_t)\tilde{P}(s_{t+2}|s_{t+1}, a_{t+1}) \prod_i P(a^i_{t+2}|s_{t+1}, c_{t+1})$$
$$\prod_i P(a^i_{t+1}|s_t, c_t) \prod_j P(c^j_{t+1}|s_t, c_t) \prod_j P(c^j_{t+2}|s_{t+1}, c_{t+1})\tilde{P}(g|s_{t+2}, s_{t+1}, s_t, a_{t+2}, a_{t+1}, a_t),\ z \in \mathcal{Z} \Bigg\}$$

We will call $\mathcal{M}_A$ the agent manifold. Note that these two manifolds are disjoint, since every distribution in $\mathcal{M}_G$ has per definition values equal to zero and is therefore on the boundary of the probability simplex.

Using the em-algorithm we are able to find the minimal difference between these two manifolds

$$\inf_{P\in\mathcal{M}_A, Q\in\mathcal{M}_G} D(Q \parallel P).$$

Therefore it results in the distribution $P \in \mathcal{M}_A$ that is closest to achieving the goal. The algorithm works by iteratively projecting to $\mathcal{M}_G$ with an e-projection, meaning minimizing the KL-divergence with respect to the first argument, and then projecting to $\mathcal{M}_A$ with an m-projection, defined by minimizing the KL-divergence with respect to the second argument. A sketch of this process is depicted in Figure 8.

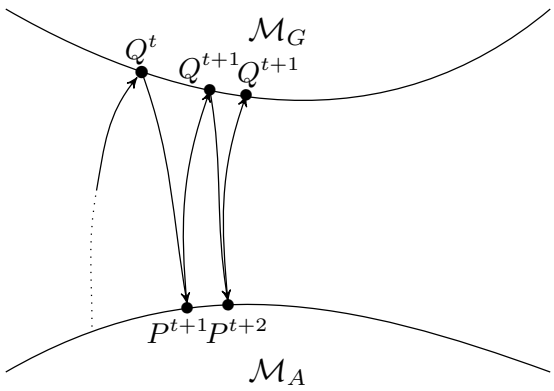


Figure 8: Sketch of the em-algorithm.

Now we take $\bar{P}^0$ as an arbitrary initial distribution as described above and project this to $\mathcal{M}_G$ via an e-projection

$$Q^0 = \arg\inf_{Q \in \mathcal{M}_G} D(Q \parallel \bar{P}^0).$$

Then we perform an m-projection to $\mathcal{M}_A$

$$P^1 = \arg\inf_{P \in \mathcal{M}_A} D(Q^0 \parallel P).$$

Repeating this leads to

$$Q^t = \arg\inf_{Q \in \mathcal{M}_G} D(Q \parallel P^t) \qquad P^{t+1} = \arg\inf_{P \in \mathcal{M}_A} D(Q^t \parallel P).$$

This em-algorithm is guaranteed to converge, but might converge towards a local minimum [5], as discussed in Section 5.3 of [13].

Now it remains to show what the projections are in our case. We will start with the e-projection. Projecting $P \in \mathcal{M}_A$ onto a linear family w.r.t. the first variable is well known and can be performed in the following way

$$\begin{aligned} \arg\min_{Q \in \mathcal{M}_G} D(Q \parallel P) = Q^\star \\ Q^\star(z) = P(z) \cdot \frac{Q(g)}{P(g)} \end{aligned} \tag{1}$$

Note that $P \in \mathcal{M}_A$ is strictly positive and that this definition is therefore well defined.

The m-projection can be performed as follows.

$$\begin{aligned} \arg\min_{P \in \mathcal{M}_A} D(Q \parallel P) =& P^\star \\ P^\star(z) =& \bar{P}(s_t, c_t, a_t)\tilde{P}(s_{t+1}|s_t, a_t)\tilde{P}(s_{t+2}|s_{t+1}, a_{t+1}) \prod_i Q(a^i_{t+1}|s_t, c_t) \prod_i Q(a^i_{t+2}|s_{t+1}, c_{t+1}) \\ & \prod_j Q(c^j_{t+1}|s_t, c_t) \prod_j Q(c^j_{t+2}|s_{t+1}, c_{t+1}) \tilde{P}(g|s_{t+2}, s_{t+1}, s_t, a_{t+2}, a_{t+1}, a_t)\}. \end{aligned} \tag{2}$$

The proofs for both projections can be found in the Appendix. This algorithm is equivalent to the EM-algorithm used in statistics, see Section 8.1 in [3] or Section 5.3 in [13].

3 Measures of the Information Flow

In this section we will define the different measures that we apply to the distribution calculated with the em-algorithm. These are information theoretic measures that use the KL-divergence to calculate the difference between the original distribution and a split distribution. This split distribution is the one that is closest to the original distribution without having the connection that we want to measure. The KL-divergence and some of its properties are discussed in Appendix 6.1.

Definition 1. Let $M \subset P^{\circ}(\mathcal{Y} \times \mathcal{Y})$ be a set of probability distributions corresponding to a split system. Then we minimize the KL-divergence between M and the initial distribution P to quantify the strength of the connections missing in the split system

$$\inf_{Q \in M} D(P \parallel Q) = \sum_{y_t, y_{t+1}} P(y_t, y_{t+1}) \, log \, \frac{P(y_t, y_{t+1})}{Q(y_t, y_{t+1})}.$$

Therefore these measures can be seen as complexity measures quantifying how much the whole distribution differs from one divided into parts.

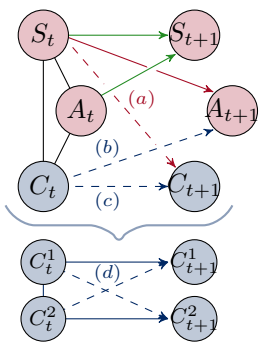
In the next subsections we will present the different measures. Except for Φ_{Syn} every measure has a closed form solution. The proofs can be found in the appendix. Additionally every measure is shown in a figure in the same section. The respective connection quantified by a measure is drawn as a dashed connection and marked with a letter corresponding to the measure.

3.1 Information Flow regarding the controller

Here we discuss measures that quantify the information flow related to the controller of the agent. The first measure restricts itself only to the controller nodes and can be seen in the context of the Integrated Information Theory of consciousness [27]. This theory was discussed in the introduction. It aims at measuring the strength of the connections among different nodes among different points in time, in other words the connections that integrate the information. Since every influence on C_{t+1} is known in our setting, we are able to use the measure Φ_T we proposed in [21]. This measure is defined in the following way:

$$\Phi_T = \sum_{y_t, y_{t+1}} P(y_t, y_{t+1}) \, log \left(\frac{\prod_j P(c^j_{t+1} | s_t, c_t)}{\prod_j P(c^j_{t+1} | s_t, c^j_t)} \right) = \sum_j I(C^j_{t+1}; C^{I \setminus \{j\}}_t | C^j_t, S_t)$$

and depicted as (d) in Figure 9. In the definition above, $I(C^j_{t+1}; C^{I \setminus \{j\}}_t | C^j_t, S_t)$ denotes the conditional mutual information, described in Definition 5, and $I \setminus \{j\}$ is the set of indices of controller nodes without j. For two controller nodes and $j = 2$ this would be $\{1, 2\} \setminus \{2\} = \{1\}$.



(a) $\Psi_{SI} = \sum_j I(C^j_{t+1}; S_t | C_t)$

(b) $\Psi_C = \sum_i I(A^i_{t+1}; C_t | S_t)$

(c) $\Psi_M = \sum_j I(C^j_{t+1}; C_t, S_t)$

(d) $\Phi_T = \sum_j I(C^j_{t+1}; C^{I \setminus j}_t | C^j_t, S_t)$

Figure 9: Calculation of the measures that analyze the information flow regarding the controller.

This value is upper bounded by the mutual information between C_t and C_{t+1}. In this case we remove the connection between the controller nodes entirely. If this value is zero, then there is no information retained

in the controller between different points in time and therefore we will call this value "memory" and denote it by Ψ_M.

$$\Phi_T \leq \Psi_M = \sum_{y_t, y_{t+1}} P(y_t, y_{t+1}) \, log \left(\frac{\prod_j P(c^j_{t+1}|s_t, c_t)}{\prod_j P(c^j_{t+1}|s_t)} \right) = \sum_j I(C^j_{t+1}; C_t|S_t)$$

These two measures only consider the controller, but since we are looking at an embodied agent, we also want to measure how much of the information processed in the controller has an actual impact on the behavior of the agent. We will term the measure quantifying the strength of the impact of the controller on the actuator "control", Ψ_C.

$$\Psi_C = \sum_{y_t, y_{t+1}} P(y_t, y_{t+1}) \, log \left(\frac{\prod_i P(a^i_{t+1}|s_t, c_t)}{\prod_i P(a^i_{t+1}|s_t)} \right) = \sum_i I(A^i_{t+1}; C_t|S_t)$$

The commands the controller sends to the actuator should be based on the information received from the sensors. Therefore we will additionally calculate the strength of the information flow from the sensor to the controller nodes. The smaller this value is, the more likely it is that the controller converged to a general strategy and performs this blindly without including the information from the sensors. We will call this "sensory information", Ψ_{SI}.

$$\Psi_{SI} = \sum_{y_t, y_{t+1}} P(y_t, y_{t+1}) \, log \left(\frac{\prod_j P(c^j_{t+1}|s_t, c_t)}{\prod_j P(c^j_{t+1}|c_t)} \right) = \sum_j I(C^j_{t+1}; S_t|C_t)$$

These measures are shown in Figure 9.

3.2 Reactive Control

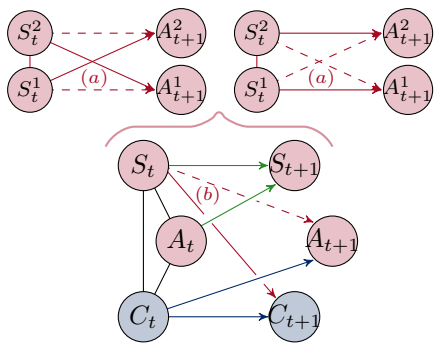
Reactive control describes a direct stimuli response, meaning that the sensors send their unprocessed information directly to the actuators. We are measuring this by the value Ψ_R. The corresponding split distribution results from removing the connection between S_t and A_{t+1}

$$\Psi_R = \sum_{y_t, y_{t+1}} P(y_t, y_{t+1}) \, log \left(\frac{\prod_i P(a^i_{t+1}|s_t, c_t)}{\prod_i P(a^i_{t+1}|c_t)} \right) = \sum_i I(A^i_{t+1}; S_t|C_t).$$

We will calculate a second measure regarding this connection. Additionally, we want to quantify how important the integration of data from different sensors is for the actuators. Therefore we will calculate two values with two different split systems. In the first split system A^1_{t+1} only depends on S^1_t and A^2_{t+1} depends only on S^2_t. There is no reason, why we would identify A^1 with S^1 and not S^2 and therefore we will also calculate a value using a split system in which A^1_{t+1} only depends on S^2_t and A^2_{t+1} only depends on S^1_t.

$$\begin{aligned}\Psi_{MSI} = min \Bigg\{ & \sum_{y_t, y_{t+1}} P(y_t, y_{t+1}) \, log \left(\frac{P(a^1_{t+1}|s_t, c_t) P(a^1_{t+1}|s_t, c_t)}{P(a^1_{t+1}|s^2_t, c_t) P(a^2_{t+1}|s^1_t, c_t)} \right), \\ & \sum_{y_t, y_{t+1}} P(y_t, y_{t+1}) \, log \left(\frac{P(a^1_{t+1}|s_t, c_t) P(a^1_{t+1}|s_t, c_t)}{P(a^1_{t+1}|s^1_t, c_t) P(a^2_{t+1}|s^2_t, c_t)} \right) \Bigg\} \\ = min \left\{ I(A^1_{t+1}; S^1_t|C_t, S^2_t) + I(A^2_{t+1}; S^2_t|C_t, S^1_t), \; I(A^1_{t+1}; S^2_t|C_t, S^1_t) + I(A^2_{t+1}; S^1_t|C_t, S^2_t) \right\}\end{aligned}$$

We will call this measure Ψ_{MSI}, which stands for multisensory integration and its definition is sketched in Figure 10.



(a) $\Psi_{MSI} = min\left\{I(A_{t+1}^1; S_t^1|C_t, S_t^2) + I(A_{t+1}^2; S_t^2|C_t, S_t^1), \right.$
$\left. I(A_{t+1}^1; S_t^2|C_t, S_t^1) + I(A_{t+1}^2; S_t^1|C_t, S_t^2)\right\}$

(b) $\Psi_R = \sum_i I(A_{t+1}^i; S_t|C_t)$

Figure 10: Calculating the measures regarding reactive control.

3.3 Morphological Computation

The concept of Morphological Computation was renamed in [16] to Morphological Intelligence and characterized in Definition 1.1. as follows

> "Morphological Intelligence is the reduction of computational cost for the brain (or controller) resulting from the exploitation of the morphology and its interaction with the environment."

The distribution $\tilde{P}(S_{t+1}|S_t, A_t)$ describes the influence the agent has on itself through the environment. Hence this distribution is dependent on the environment and the morphology of the agent, which is given as the length of the sensors. In [17] the authors define two measures for Morphological Computation that depend on $\tilde{P}(S_{t+1}|S_t, A_t)$. We are able to quantify the effect of the action on the next sensory state by calculating

$$\sum_{y_t, y_{t+1}} P(y_t, y_{t+1}) \, log \left(\frac{P(s_{t+1}|s_t, a_t)}{P(s_{t+1}|s_t)} \right) = I(S_{t+1}; A_t|S_t).$$

This quantifies the amount of control an agent has and therefore the corresponding measure for morphological computation has to be inverted. In order to assure that this measure is positive, we also normalize it

$$\Psi_A = 1 - \frac{1}{log|\mathcal{S}|} \sum_{y_t, y_{t+1}} P(y_t, y_{t+1}) \, log \left(\frac{P(s_{t+1}|s_t, a_t)}{P(s_{t+1}|s_t)} \right) = 1 - \frac{1}{log|\mathcal{S}|} I(S_{t+1}; A_t|S_t).$$

Note that

$$\begin{aligned} \sum_{y_t, y_{t+1}} P(y_t, y_{t+1}) \, log \left(\frac{P(s_{t+1}|s_t, a_t)}{P(s_{t+1}|s_t)} \right) &\leq - \sum_{y_t, y_{t+1}} P(y_t, y_{t+1}) \, log \, P(s_{t+1}|s_t) \\ &= - \sum_{y_t, y_{t+1}} P(y_t, y_{t+1}) \, log \frac{P(s_{t+1}|s_t)}{\frac{1}{|S|}} + log \frac{1}{|S|} \leq log \frac{1}{|\mathcal{S}|} \end{aligned}$$

The second measure quantifies the strength of the influence of the past sensory input on the next sensory input as

$$\Psi_S = \sum_{y_t, y_{t+1}} P(y_t, y_{t+1}) \, log \left(\frac{P(s_{t+1}|s_t, a_t)}{P(s_{t+1}|a_t)} \right) = I(S_{t+1}; S_t|A_t)$$

which corresponds to $ASOC_W$ defined in [16] in Definition 3.1.3.

The third measure for Morphological Computation is called synergistic information Ψ_{Syn}, was proposed in [18] and is conceptually different from the other measures. The split distribution Q does not simply remove one connection, but is defined in a way, that the two way interactions among the three parts A_t, S_t and S_{t+1} stay fixed to the two way interaction of the initial distribution. This means that

$$P(A_t, S_t) = Q(A_t, S_t), \quad P(A_t, S_{t+1}) = Q(A_t, S_{t+1}) \text{ and } P(S_t, S_{t+1}) = Q(S_t, S_{t+1}) \tag{3}$$

hold. Since we do not want to make any further assumptions on the structure of the split distribution, we calculate Q by maximizing the entropy under the constraints (3). By doing so we remove the synergistic interactions among A_t and S_t on S_{t+1}. This can be done by using an iterative scaling algorithm that cycles through the constraints and projects in each step to a set of distributions satisfying one constraint. There are various versions of the iterative scaling algorithm. For a more detailed description see [11].

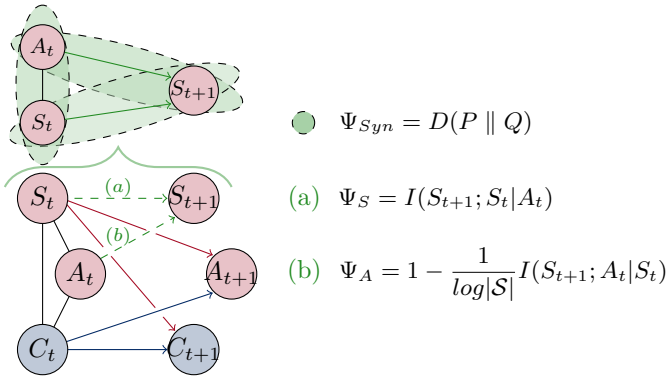


Figure 11: Calculating measures for Morphological Computation.

3.4 Predictive Information

The measures discussed in this section assess how useful information from the past is to predict the next sensor state. This is done by calculating the mutual information between the past and present sensory state. In Section 3 of [19] this was introduced as a measure for the complexity of ensembles of patterns and named "effective measure complexity". Bialek et al. further analyze this measure in [9] and name it "predictive information". The predictive information in the sensor space of robots was used in [7] as an objective function for self-organization and there if proves effective in a simulated chain of passively coupled two wheeled robots in a maze.

The measure for predictive information is therefore only defined on the sensor states S_{t+1}, S_t as

$$\Psi_{PI} = \sum_{x_t, x_{t+1}} P(y_t, y_{t+1}) \, log \left(\frac{P(s_{t+1}|s_t)}{P(s_{t+1})} \right) = I(S_{t+1}; S_t).$$

Additionally we are able to include the last actuator value to calculate how predictable the next sensory input is given the knowledge of the last sensor states and the initiated action:

$$\Psi_{SA} = \sum_{x_t, x_{t+1}} P(y_t, y_{t+1}) \, log \left(\frac{P(s_{t+1}|s_t, a_t)}{P(s_{t+1})} \right) = I(S_{t+1}; S_t, A_t).$$

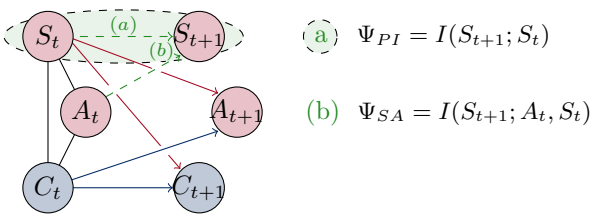


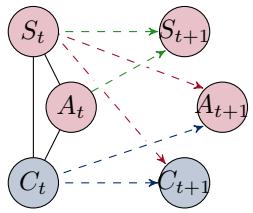
Figure 12: Calculating measures related to predictive information.

If this measure is zero, then the next sensory input does not depend on the past sensor state or the past action. In this case we would not be able to optimize the behavior of the agent since it has no impact on its survival.

3.5 Total Information Flow

Finally, we will also calculate the total information flow by removing all the connections between Y_t and Y_{t+1} in the split system.

$$\Psi_{TIF} = \sum_{y_t, y_{t+1}} P(y_t, y_{t+1}) \, log \left(\frac{P(s_{t+1}|s_t, a_t) \prod_i P(a^i_{t+1}|s_t, c_t) \prod_j P(c^j_{t+1}|s_t, c_t)}{P(s_{t+1}) \prod_i P(a^i_{t+1}) \prod_j P(c^j_{t+1})} \right)$$
$$= I(S_{t+1}; S_t, A_t) + I(A_{t+1}; S_t, C_t) + I(C_{t+1}; C_t, S_t)$$



$$\Psi_{IIF} = I(S_{t+1}; S_t, A_t) + I(A_{t+1}; S_t, C_t) + I(C_{t+1}; C_t, S_t)$$

Figure 13: Calculating the total information flow.

This is an upper bound for the previously discussed measures, except for Ψ_{Syn} and Ψ_A.

4 Results

In this section we will present the results of our experiments. The length of the sensors are varied from 0.5 to 2 in steps of 0.25. We took 50 random input distributions $\bar{P}$ and for each $\bar{P}$ we used the em-algorithm three times to find three different full distributions.

(1) $P(y_t, y_{t+1}) = \bar{P}(s_t, c_t, a_t)\tilde{P}(s_{t+1}|s_t, a_t) \prod_i P(a^i_{t+1}|s_t, c_t) \prod_j P(c^j_{t+1}|s_t, c_t)$

(2) $P(y_t, y_{t+1}) = \bar{P}(s_t, c_t, a_t)\tilde{P}(s_{t+1}|s_t, a_t) \prod_i P(a^i_{t+1}|c_t) \prod_j P(c^j_{t+1}|s_t, c_t)$

(3) $P(y_t, y_{t+1}) = \bar{P}(s_t, c_t, a_t)\tilde{P}(s_{t+1}|s_t, a_t) \prod_i P(a^i_{t+1}|s_t, c_t) \prod_j P(c^j_{t+1}|s_t, c_t^{J \setminus \{j\}})$

The distribution (1) is the full distribution described in the previous sections as shown in Figure 14. Additionally, we analyzed the full distribution (2), which does not have a connection between S_t and A_{t+1} and therefore no reactive control. This is an additional way to analyze the information flow by directly manipulating the architecture of the agents. Furthermore, the distribution (3) has no integrated information in the controller to begin with. We will draw the graphs corresponding to (1) in the following figures in the first column and as continuous lines, the graphs corresponding to (2) in the second column as dashed lines and the graph corresponding to (3) as dotted lines in the last column.

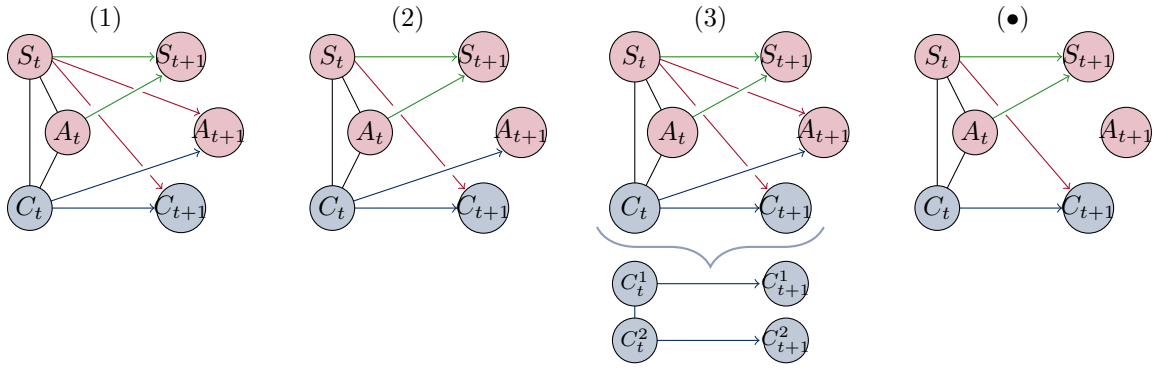


Figure 14: The three different full distributions (1), (2), (3) and the distribution that indicated a lower bound (•) on the performance.

The solid black line in the graphs in Figure 15 show the measures calculated for a full distribution defined as

$$P(x_t, x_{t+1}) = \bar{P}(s_t, c_t, a_t)\tilde{P}(s_{t+1}|s_t, a_t)\prod_i P(a_{t+1}^i)\prod_j P(c_{t+1}^j|s_t, c_t^j).$$

The corresponding graph can be seen on the right in Figure 14. In this case the controller or the sensory input have no influence on the action. The em-algorithm optimizes the distribution $P(A_t)$ regardless of any sensory input. An agent that processes the information from the environment should perform as least as good as the agent ignoring the sensory information and therefore we will use the measure resulting from this agent as a lower bound to compare the performance of the agents to.

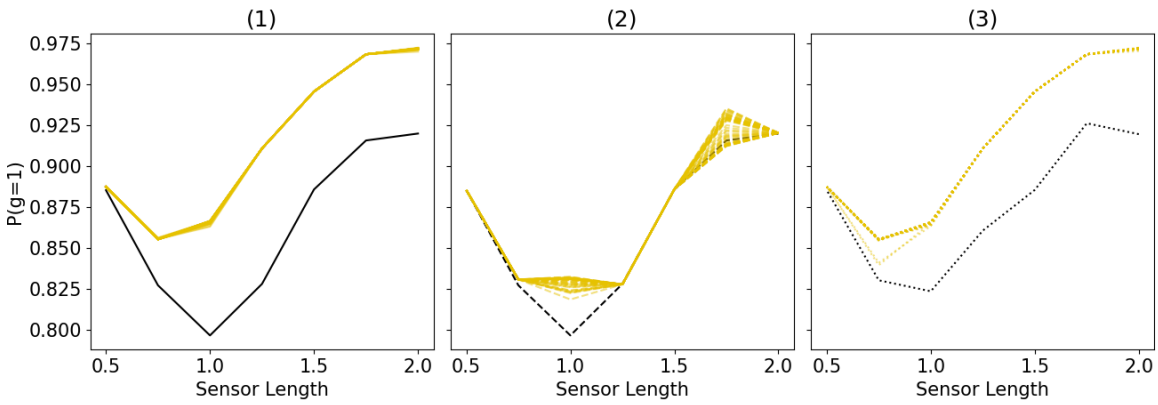


Figure 15: The probability of the agents reaching the goal $P(g = 1)$.

Now we are going to present our results. The Figure 15 shows the probability with which the agents survive $P(g = 1)$. In the first column we are able to observe that agents that rely on the information perform in most cases significantly better than the lower bound. This does not hold in the case of a sensor length of 0.5. If the sensors are that small, then the information that a sensor has touched a wall has no time to pass through the agent before it dies. This phenomenon is even stronger in the second column, where we have no direct connection between the sensors and the actuators. There are only two sensor length in which the additional information makes a difference: 1 and 1.75. We will see that these points also show a significantly different behavior for the measures presented below. In (3) the graphs behave similar to the graphs in (1) with the exception, that we have two cases that perform worse than the rest at 0.75. An explanation for this is that the em-algorithm converged to a local minimum in these cases.

The first measures we are going to look at are the once regarding the controller depicted in Figure 16. All of these measures have a significant decrease for the sensor length 0.5, because there the sensors are too short for the information flowing through the controller to make a difference. Hence the information flow does not get maximized by the em-algorithm.

In the first row we see the measure quantifying how important the information flow from the sensors to the controller is. For the distributions (1) the maximal value is at either 0.75 or 1 and decreases for longer sensors. We will discuss this decrease in relation to the other measures below. The graphs for Ψ_{ID} in the case (2) show a different behavior. There we have two spikes, one at 1.0 and one at 1.75. These are exactly the two sensor length for which the agents perform better than the lower bound as shown in Figure 15. For the full distributions without integrated information in (3) the sensory information going to the controller has no real impact on the behavior.

The importance of the commands send form the controller to the actuator measured by Ψ_C is depicted in the second row. This measure behaves similarly to Ψ_{ID} for (1) and (2). In (3) we have a spike at 0.75 for two of the distributions. These are the once that correspond most likely to a local minimum mentioned above in the context of Figure 15.

The last two rows show Ψ_M and Φ_T. These measures behave very similar, but notice that Ψ_M is an upper bound for Φ_T. We will see the importance of the integrated information in the discussion of Figure 17. Here we want to point out a different observation. If we would only look at Φ_T to measure the importance of the information flow in the controller, we would conclude that there are agents for which the information flow through the controller for the sensor length of 1 or 0.75 is not important. But if we include Ψ_{SI} and Ψ_C into our analysis, we see that although the measures Ψ_M and Φ_T might be low, they still have a significant impact on the behavior of the agent. Therefore the importance of the information flow in the controller of an embodied agent depends additionally on the information flowing to and from the controller. This is the 1. statement in the introduction.

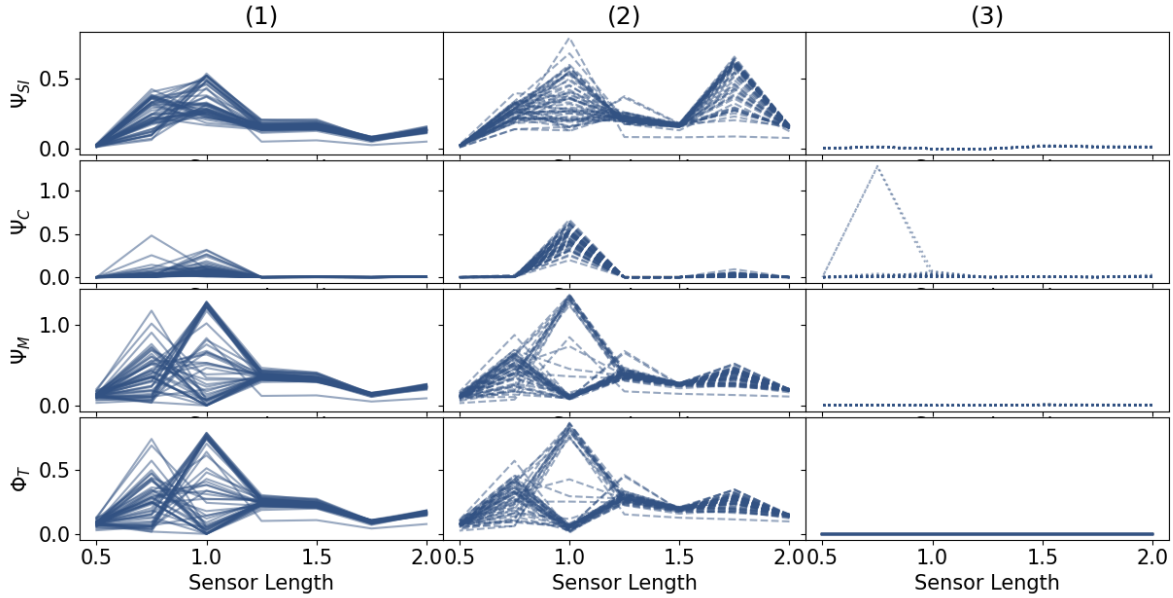


Figure 16: The measures regarding the controller.

The measures shown in Figure 17 examine the information flow from the sensors directly to the actuators. It is not surprising that the importance of this flow increases monotonically with the length of the sensors, since the predictability and therefore the informational value of the sensors increases. This can be seen in Figure 19. The second distribution (2) has per definition $\Psi_R = 0$ and $\Psi_{MSI} = 0$. Now we compare the first and the third column. The measure for multisensori integration Ψ_{MSI} shows the same general behavior as Ψ_R, but with lower values. On the other hand, we need to take into account, that the direct connection

from the sensors to the actuators have a higher impact on the behavior of the agents, since the influence from the controller is low, as discussed above.

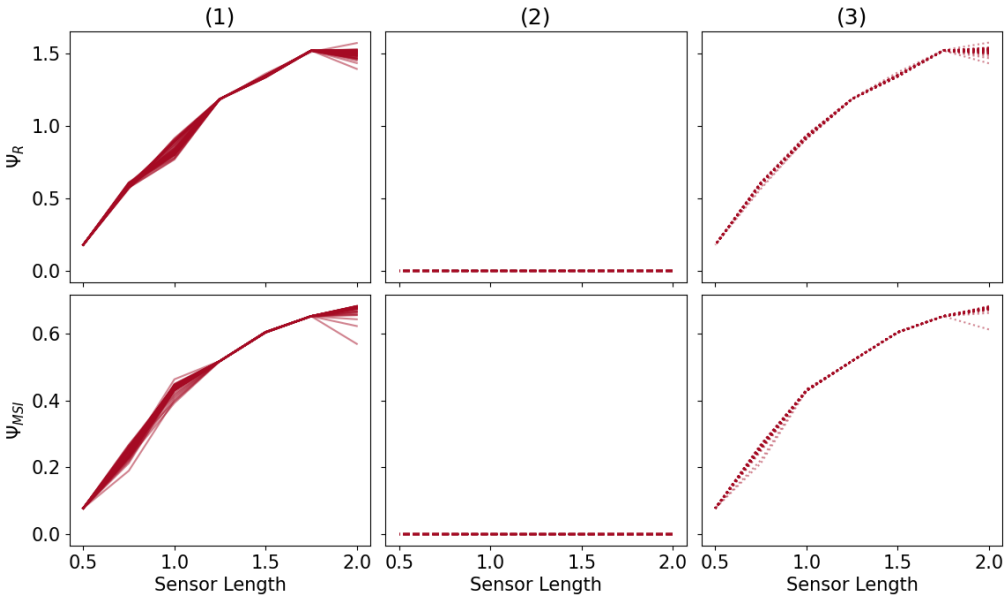


Figure 17: The measures corresponding to reactive control.

The measures regarding Morphological Computation and predictive information only depend on the sampled distribution $\tilde{P}(S_{t+1}|S_t, A_t)$ and therefore we do not need to differentiate between the cases $(1), (2), (3)$ or different initial distributions. In Figure 18 we see that Ψ_{Syn} and Ψ_A behave very similar. The smallest value is at a sensor length 1 and increases in both directions. The measure Ψ_{Syn} quantifies the synergistic information of (A_t, S_t) on S_{t+1}, meaning how much S_{t+1} depends on the interplay between (A_t, S_t). We see that considering the combination of A^t and S^t becomes more important when the sensors are very small or very long. For the sensor length of 1 the actuator A_t alone has a high impact on the next sensor state, since that is the minimum of the measure Ψ_A. Now we are going to compare the measures related to the controller in Figure 16 with the first two measures for Morphological Computation. We see that the sensor length where $\Phi_T, \Psi_{ID}, \Psi_C$ and Ψ_M achieve the highest values mostly at a sensor length of 1 and therefore we are able to observe an asymmetry between the Morphological Computation and integrated information described in the 2. statement. The less the agent is able to rely on the controller, the more important Morphological Computation becomes.

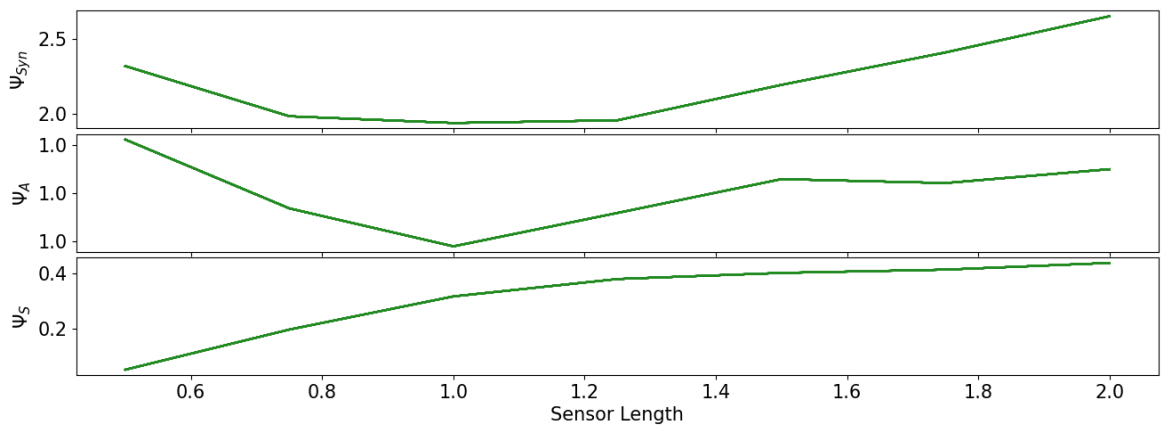


Figure 18: The measures quantifying Morphological Computation.

The measure Φ_S shows a different behavior for a smaller sensor length. This measure is defined as the mutual information between S^{t+1} and S^t given A^t and therefore it measures the influence of the last sensory input on the present sensory input given a controller input. Hence this measure relies more on the length of the sensors than Ψ_{Syn} and Ψ_A. It is closely related to the measure for predictive information shown in Figure 19.

In Figure 19 we see that the measures monotonically increase with the length of the sensors. This is an expected outcome, because the longer the sensors are, the more the agents knows about its environment, hence the more certain the next input is.

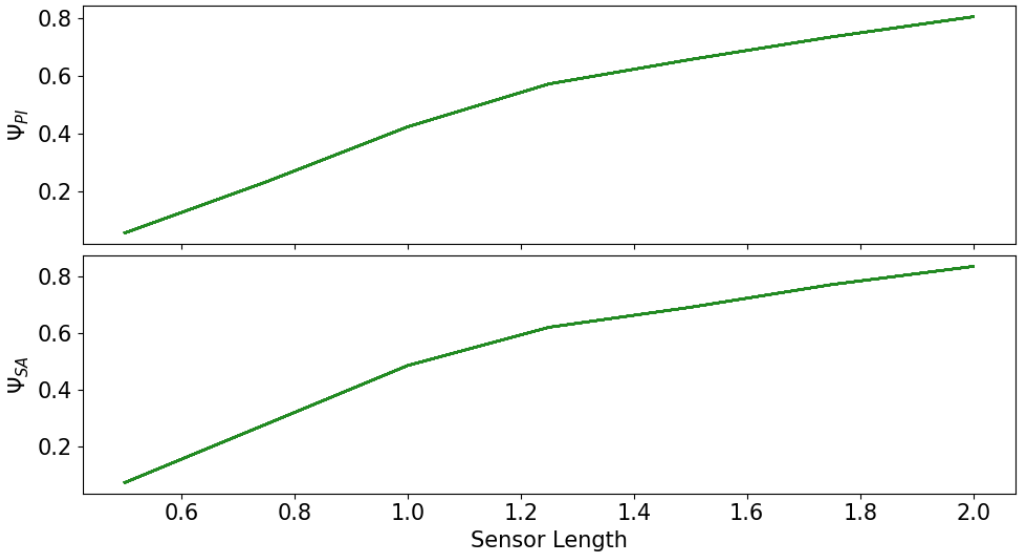


Figure 19: The measures corresponding to predictive information.

The last figure, Figure 4, shows the value of the total information flow. In the case of distribution (1) we see a spike for 1.0 corresponding to the measures regarding the controller. The total information flow in (2) depicts two spikes at the sensor length that differ from the lower bound as discussed above. The distributions without integrated information in the controller in (3) show a total information flow that behaves mostly like reactive control and Ψ_{SA}, except for the spike for the local minima at 0.75. Therefore the total information flow combined with the measures above give us a notion of what measure has the most influence on the agent.

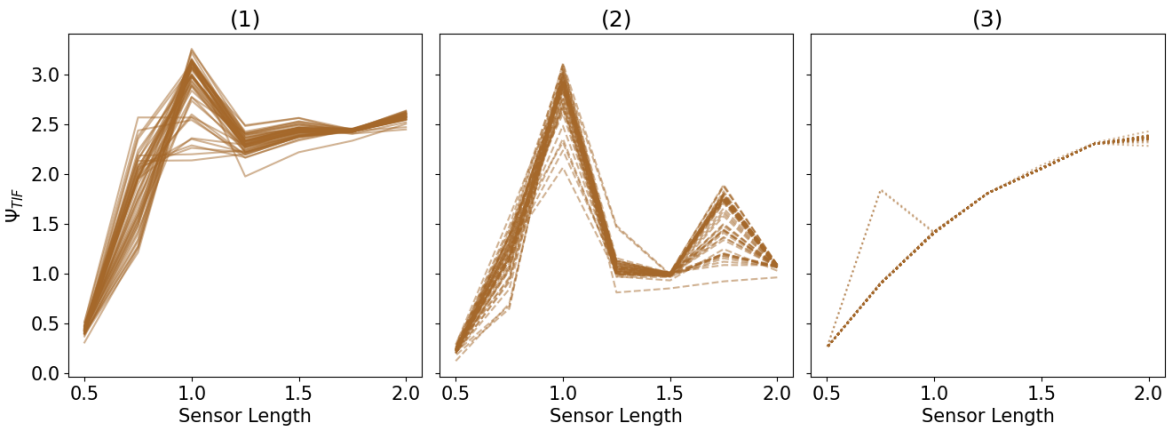


Figure 20: The measure quantifying the total information flow.

5 Discussion

In this article we combined different techniques in order to create a framework to analyze the information flow among an agents body, its controller and the environment. The main question we want to approach in this framework is how the complexity of solving a task is distributed among these different interacting parts. We demonstrate the steps in the analysis with the example of small simulated agents that are not allowed to touch the walls of a racetrack. These agents have a sufficiently simple architecture such that we are able to rigorously analyze the different information flows of the optimal behavior. Additionally, we can examine the dynamics of the information theoretic measures of an agent under changing morphological circumstance by modifying the length of the sensors.

We calculate the optimal behavior by using the concept of Planning as Inference, that allows us to model the conditional distributions $P(C_{t+1}|C_t, S_t)$ and $P(A_{t+1}|C_t, S_t)$ as latent variables. Using the information geometric em-algorithm, we are able to optimize the latent variables such that the probability of success is maximal. Here, the expectation maximization EM algorithm used in statistics is equivalent to the em-algorithm, but we chose to present the em-algorithm, because it has an intuitive geometric interpretation.

The distributions that are optimal regarding reaching a goal are then analyzed by applying twelve information theoretic measures. We use the measure Φ_T to calculate the integrated information in the controller and we demonstrate that the importance of the information flow in the controller of an embodied agent additionally depends on the information flow to and from the controller, measured by Ψ_{SI} and Ψ_C . This is our 1. statement from the introduction.

We apply three different measures for Morphological Computation Ψ_{Syn}, Ψ_A and Ψ_S and we observe that the first two measures behave similar. Comparing Ψ_{Syn}, Ψ_A to the measures regarding the controller reveals an asymmetric relationship between integrated information and Morphological Computation. If the length of the sensors are too small for the information passing through the controller to matter, then Ψ_{Syn}, Ψ_A increase. On the other hand, when the sensors are long enough (≥ 1.25) for the agent to effectively use Morphological Computation, then integrated information decreases. This is the phenomenon described in the 2. statement.

The measure Ψ_{PI} for Predictive Information shows the expected result, that predictability increases with the length of the sensors.

All in all, we present a method to completely examine the information flow among the brain, body and environment of an agent. This gives us insights into how the complexity of the task is met by the different interacting components. We will continue to develop these concepts further to be able to efficiently analyze more complicated agents and tasks.

Acknowledgement

The authors acknowledge funding by Deutsche Forschungsgemeinschaft Priority Programme “The Active Self” (SPP 2134). Additionally we thank Nathaniel Virgo for the design and implementation of the racetrack and for the introduction to the theory of Planning as Inference.

6 Appendix

6.1 Information Theory

In this section we define important information theoretic concepts, further details can be found in [10]. Let P, Q be probability distributions of random variables on the state space $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$.

Definition 2 (Entropy, Cross-Entropy)**.** In information theory the entropy H_P measures how uncertain the outcome of a random variable is

$$H_P = -\sum_{z \in \mathcal{Z}} P(z) \, log \, P(z)$$

with the convention that $0 \cdot log \, 0 = 0$.

Using a different distribution Q instead of the true distribution P leads to the cross-entropy $H_{P,Q}$

$$H_{P,Q} = -\sum_{z \in \mathcal{Z}} P(z) \, log \, Q(z).$$

The concepts of entropy and cross-entropy directly lead to the definition of the KL-divergence, also called relative entropy.

Definition 3 (KL-Divergence)**.** The Kullback-Leibler-divergence is defined as

$$D(P \parallel Q) = \sum_{z \in \mathcal{Z}} P(z) \, log \left(\frac{P(z)}{Q(z)} \right) = H_{P,Q} - H_P$$

with the conventions that $0 \cdot log \, \frac{0}{0} = 0$, $0 \cdot log \, \frac{0}{Q(z)} = 0$ and $P(z) \cdot log \, \frac{P(z)}{0} = \infty$ for $P(z) > 0$.

This measures how much the uncertainty of the random variable increases, if we use Q instead of P. The KL-divergence has the following properties:

1. $D(P \parallel Q) \geq 0$
2. $D(P \parallel Q) = 0$ if and only if $P = Q$

Proofs of these properties can be found in [10] in Theorem 2.6.3. It follows directly from the definition of the KL-divergence and property 1. that the cross-entropy is greater or equal to entropy.

The minimization of the KL-divergence with respect to the second argument is called rI-projection or m-projection. Minimizing with respect to the first argument is named I-projection or e-projection. These terms have their origin in Information Geometry, further information about the projections can be found in [12] or in Section 2.8.3 in [8].

The mutual information defined below is a special case of the KL-divergence.

Definition 4 (Mutual Information)**.** Let (X, Y) be a random vector on $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ with the distribution P. Then the mutual information is defined as the KL-divergence between $P(X, Y)$ and the product distribution $P(X)P(Y)$

$$I(X;Y) = \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} P(x, y) \, log \left(\frac{P(x, y)}{P(x)P(y)} \right).$$

In order to define the conditional mutual information, we need to have three random variables X_1, X_2, X_3 on the state spaces $\mathcal{X}_1, \mathcal{X}_2$ and $\mathcal{X}_3$ with the probability distribution P.

Definition 5 (Conditional Mutual Information)**.** The conditional mutual information of the random variables X_1 and X_2 given X_1 is defined by

$$I(X_1; X_2|X_3) = \sum_{x_1 \in \mathcal{X}_1} \sum_{x_2 \in \mathcal{X}_2} \sum_{x_3 \in \mathcal{X}_3} P(x_1, x_2, x_3) \, log \left(\frac{P(x_1, x_2|x_3)}{P(x_1|x_3)P(x_2|x_3)} \right)$$
$$\sum_{x_1 \in \mathcal{X}_1} \sum_{x_2 \in \mathcal{X}_2} \sum_{x_3 \in \mathcal{X}_3} P(x_1, x_2, x_3) \, log \left(\frac{P(x_1|x_2, x_3)}{P(x_1|x_3)} \right).$$

6.2 Proofs

Proof of Proposition 1. As defined in the document we write $y_t = (s_t, c_t, a_t) \in \mathcal{Y} = \mathcal{S} \times \mathcal{C} \times \mathcal{A}$,

$$
\begin{aligned}
&P(y_t, y_{t+1}) \\
&= \sum_{w_t, w_{t+1}} P(w_t) \cdot P(y_t|w_t) \cdot P(w_{t+1}|w_t, a_t) \prod_k P(s^k_{t+1}|w_{t+1}) \prod_i P(a^i_{t+1}|s_t, c_t) \prod_j P(c^j_{t+1}|s_t, c_t) \\
&= \prod_i P(a^i_{t+1}|s_t, c_t) \prod_j P(c^j_{t+1}|s_t, c_t) \sum_{w_t, w_{t+1}} P(w_t) \cdot P(y_t|w_t) \cdot P(w_{t+1}|w_t, a_t) \prod_k P(s^k_{t+1}|w_{t+1}) \\
&= \prod_i P(a^i_{t+1}|s_t, c_t) \prod_j P(c^j_{t+1}|s_t, c_t) \cdot \tilde{P}(y_t, s_{t+1}) \\
&= P(s_t, a_t, c_t) \cdot \prod_i P(a^i_{t+1}|s_t, c_t) \prod_j P(c^j_{t+1}|s_t, c_t) \cdot \tilde{P}(s_{t+1}|y_t)
\end{aligned}
$$

Now we take a closer look at $P(s_{t+1}|y_t)$ and show that S_{t+1} is independent of C_t given (S_t, A_t). For that we need to describe $P(y_t)$ in more detail. The graph describing the distribution is a chain graph and therefore we are able to use Section 3.2.3 in [22]. Hence there exist non-negative functions Φ_1, Φ_2, such that $P(s_t, a_t, c_t) = \Phi_2(s_t, a_t, c_t) \sum_{w_t} \Phi_1(s_t, w_t)$. Using this definition results in

$$
\begin{aligned}
\tilde{P}(s_{t+1}|y_t) &= \frac{P(s_{t+1}, y_t)}{P(y_t)} \\
&= \frac{\sum_{w_t} \Phi_1(s_t, w_t) \sum_{w_{t+1}} P(w_{t+1}|w_t, a_t) P(s_{t+1}|w_{t+1})}{\sum_{w_t} \Phi_1(s_t, w_t)} \\
&= \tilde{P}(s_{t+1}|s_t, a_t)
\end{aligned}
$$

Therefore the factorization of P can be written as

$$
P(y_t, y_{t+1}) = P(y_t) \cdot \prod_i P(a^i_{t+1}|s_t, c_t) \prod_j P(c^j_{t+1}|s_t, c_t) \cdot \tilde{P}(s_{t+1}|s_t, a_t).
$$

□

Proof of (1). In order to proof that $Q^\star$ is the e-projection of P to $\mathcal{M}_G$ we make use of the log-sum inequality and the convention that $0 \cdot log 0 = 0$. Let $Q \in \mathcal{M}_G$ then

$$
\begin{aligned}
D(Q \parallel P) &= \sum_g \sum_z Q(z) \, log \left(\frac{Q(z)}{z)} \right) \\
&\geq \sum_g \left(\sum_{x_t, x_{t+1}, x_{t+2}} Q(z) \right) log \left(\frac{\sum_{x_t, x_{t+1}, x_{t+2}} Q(z)}{\sum_{x_t, x_{t+1}, x_{t+2}} P(z)} \right) \\
&= 1 * log \left(\frac{1}{P(g)} \right) + 0 * log \left(\frac{0}{P(g)} \right) \\
&= \sum_{x_t, x_{t+1}, x_{t+2}} P(z) \frac{1}{P(g)} * log \left(\frac{1 \cdot P(z)}{P(g) P(z)} \right) \\
&+ 0 * P(z) \frac{1}{P(g)} * log \left(\frac{0 \cdot P(z)}{P(g) P(z)} \right) \\
&= D(Q^\star \parallel P)
\end{aligned}
$$

□

Proof of (2). The KL-divergence between $Q \in \mathcal{M}_A$ and $P \in \mathcal{M}_G$ can be written as follows

$$
\begin{aligned}
D(Q \parallel P) &= \sum_z Q(z)\, log\, \frac{Q(z)}{P(z)} \\
&= \sum_z Q(z)\, log\, \frac{Q(z)}{\bar{P}(s_t, c_t, a_t)\tilde{P}(s_{t+1}|s_t, a_t)\tilde{P}(s_{t+2}|s_{t+1}, a_{t+1})\prod_i P(a^i_{t+1}|s_t, c_t)} \\
&\cdot \frac{1}{\prod_i P(a^i_{t+2}|s_{t+1}, c_{t+1})\prod_j P(c^j_{t+1}|s_t, c_t)\prod_j P(c^j_{t+2}|s_{t+1}, c_{t+1})\tilde{P}(g|s_{t+2}, s_{t+1}, s_t, a_{t+2}, a_{t+1}, a_t)} \\
&= \sum_z Q(z)\, log\, \frac{Q(z)}{\bar{P}(s_t, c_t, a_t)\tilde{P}(s_{t+1}|s_t, a_t)\tilde{P}(s_{t+2}|s_{t+1}, a_{t+1})\tilde{P}(g|s_{t+2}, s_{t+1}, s_t, a_{t+2}, a_{t+1}, a_t)} \\
&+ \sum_z Q(z)\, log\, \frac{1}{\prod_i P(a^i_{t+1}|s_t, c_t)} \\
&+ \sum_z Q(z)\, log\, \frac{1}{\prod_i P(a^i_{t+2}|s_{t+1}, c_{t+1})} \\
&+ \sum_z Q(z)\, log\, \frac{1}{\prod_j P(c^j_{t+1}|s_t, c_t)} \\
&+ \sum_z Q(z)\, log\, \frac{1}{\prod_j P(c^j_{t+2}|s_{t+1}, c_{t+1})}
\end{aligned}
$$

The last four sums are called cross-entropies and since cross-entropy is greater or equal to entropy, we gain the following inequality.

$$
\begin{aligned}
D(Q \parallel P) \geq &\sum_z Q(z)\, log\, \frac{Q(z)}{\bar{P}(s_t, c_t, a_t)\tilde{P}(s_{t+1}|s_t, a_t)\tilde{P}(s_{t+2}|s_{t+1}, a_{t+1})\tilde{P}(g|s_{t+2}, s_{t+1}, s_t, a_{t+2}, a_{t+1}, a_t)} \\
&+ \sum_z Q(z)\, log\, \frac{1}{\prod_i Q(a^i_{t+1}|s_t, c_t)} \\
&+ \sum_z Q(z)\, log\, \frac{1}{\prod_i Q(a^i_{t+2}|s_{t+1}, c_{t+1})} \\
&+ \sum_z Q(z)\, log\, \frac{1}{\prod_j Q(c^j_{t+1}|s_t, c_t)} \\
&+ \sum_z Q(z)\, log\, \frac{1}{\prod_j Q(c^j_{t+2}|s_{t+1}, c_{t+1})} \\
&= D(Q \parallel P^\star)
\end{aligned}
$$

□

Proofs of the measures in 3. Every proof can be done by applying the same technique. We will use the factorization of the elements in the split system. Then we are able to replace the cross-entropies with

entropies to find a lower bound.

$$
\begin{aligned}
\Phi_T &= \inf_{Q \in M} D(P \parallel Q) \\
D(P \parallel Q) &= \sum_{y_t, y_{t+1}} P(y_t, y_{t+1}) \, log \, \frac{\bar{P}(s_t, a_t, c_t)\tilde{P}(s_{t+1}|s_t, a_t) \prod_i P(a^i_{t+1}|s_t, a_t) \prod_j P(c^j_{t+1}|s_t, a_t)}{Q(s_t, a_t, c_t)Q(s_{t+1}|s_t, a_t) \prod_i Q(a^i_{t+1}|s_t, a_t) \prod_j Q(c^j_{t+1}|s_t, a_t)} \\
&= \sum_{y_t, y_{t+1}} P(y_t, y_{t+1}) \, log \, P(y_t, y_{t+1}) + \sum_{y_t, y_{t+1}} P(y_t, y_{t+1}) \, log \, \frac{1}{Q(s_t, a_t, c_t)} \\
&+ \sum_{y_t, y_{t+1}} P(y_t, y_{t+1}) \, log \, \frac{1}{Q(s_{t+1}|s_t, a_t)} + \sum_{y_t, y_{t+1}} P(y_t, y_{t+1}) \, log \, \frac{1}{\prod_i Q(a^i_{t+1}|s_t, a_t)} \\
&+ \sum_{y_t, y_{t+1}} P(y_t, y_{t+1}) \, log \, \frac{1}{\prod_j Q(c^j_{t+1}|s_t, a_t)} \\
&\geq \sum_{y_t, y_{t+1}} P(y_t, y_{t+1}) \, log \, P(y_t, y_{t+1}) + \sum_{y_t, y_{t+1}} P(y_t, y_{t+1}) \, log \, \frac{1}{P(s_t, a_t, c_t)} \\
&+ \sum_{y_t, y_{t+1}} P(y_t, y_{t+1}) \, log \, \frac{1}{P(s_{t+1}|s_t, a_t)} + \sum_{y_t, y_{t+1}} P(y_t, y_{t+1}) \, log \, \frac{1}{\prod_i P(a^i_{t+1}|s_t, a_t)} \\
&+ \sum_{y_t, y_{t+1}} P(y_t, y_{t+1}) \, log \, \frac{1}{\prod_j P(c^j_{t+1}|s_t, a_t)} \\
&= \sum_{y_t, y_{t+1}} P(y_t, y_{t+1}) \, log \, \frac{\prod_j P(c^j_{t+1}|s_t, a_t)}{\prod_j P(c^j_{t+1}|s_t, a_t)}
\end{aligned}
$$

This can be done with every presented measure except for Ψ_{Syn}. □

References

[1] L. Albantakis, A. Hintze, C. Koch, C. Adami, and G. Tononi. "Evolution of Integrated Causal Structures in Animats Exposed to Environments of Increasing Complexity". In: *PLOS Computational Biology* 10.12 (Dec. 2014), pp. 1–19. DOI: 10.1371/journal.pcbi.1003966. URL: https://doi.org/10.1371/journal.pcbi.1003966.

[2] L. Albantakis and G. Tononi. "The Intrinsic Cause-Effect Power of Discrete Dynamical Systems—From Elementary Cellular Automata to Adapting Animats". In: *Entropy* (2015).

[3] S. Amari. *Information Geometry and Its Applications*. Springer Japan, 2016.

[4] S. Amari. "Information geometry of Boltzmann machines". In: *IEEE Transactions on Neural Networks* (1992).

[5] S. Amari. "Information Geometry of the EM and em Algorithms for Neural Networks". In: *Neural Networks* (1995).

[6] Hagai Attias. "Planning by Probabilistic Inference". In: *Proc. of the 9th Int. Workshop on Artificial Intelligence and Statistics*. 2003.

[7] N. Ay, N. Bertschinger, R. Der, F. Güttler, and E. Olbricht. "Predictive information and explorative behavior of autonomous robots". In: *Eur. Phys. J. B* (2008).

[8] N. Ay, J. Jost, H. V. Lê, and L. Schwachhöfer. *Information Geometry*. Springer International Publishing, 2017.

[9] W. Bialek, I. Nemenman, and N. Tishby. "Predictability, Complexity, and Learning". In: *Neural Comput.* 13.11 (Nov. 2001), pp. 2409–2463. ISSN: 0899-7667. DOI: 10.1162/089976601753195969. URL: https://doi.org/10.1162/089976601753195969.

[10] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. John Wiley & Sons, Inc., 2006.

[11] I. Csiszár. "A geometric interpretation of Darroch and Ratcliff's Generalized Iterative Scaling". In: *The Annals of Statistics* (1989).

[12] I. Csiszár and F. Matúš. "Information Projections Revisited". In: *IEEE Transactions on Information Theory* (2003).

[13] I. Csiszár and P. C. Shields. *Information Theory and Statistics: A Tutorial*. Now Publishers Inc, 2004.

[14] I. Csiszár and G. Tusnády. "Information geometry and alternating minimization procedures". In: *Statistics and Decisions* (1984).

[15] J. A. Edlund, N. Chaumont, A. Hintze, C. Koch, G. Tononi, and C. Adami. "Integrated Information Increases with Fitness in the Evolution of Animats". In: *PLoS Computational Biology* (2011).

[16] K. Ghazi-Zahedi. *Morphological Intelligence*. Springer, 2019.

[17] K. Ghazi-Zahedi and N. Ay. "Quantifying Morphological Computation". In: *Entropy* (2013).

[18] K. Ghazi-Zahedi, C. Langer, and N. Ay. "Morphological Computation: Synergy of Body and Brain". In: *Entropy* 19.9 (2017).

[19] P. Grassberger. "Toward a quantitative theory of self-generated complexity". In: *International Journal of Theoretical Physics* 25 (Jan. 1986), pp. 907–938. DOI: 10.1007/BF00668821.

[20] A. Klyubin, D. Polani, and C. Nehaniv. "Representations of Space and Time in the Maximization of Information Flow in the Perception-Action Loop". In: *Neural computation* 19 (Oct. 2007), pp. 2387–432. DOI: 10.1162/neco.2007.19.9.2387.

[21] C. Langer and N. Ay. "Complexity as Causal Information Integration". In: *Entropy* 22.10 (2020).

[22] S. L. Lauritzen. *Graphical Models*. Clarendon Press, 1996.

[23] M. Lungarella, T. Pegors, D. Bulwinkle, and O. Sporns. "Methods for quantifying the informational structure of sensory and motor data". In: *Neuroinform.* (2005).

[24] M. Lungarella and O. Sporns. "Mapping Information Flow in Sensorimotor Networks". In: *PLOS Computational Biology* 2.10 (Oct. 2006), pp. 1–12. DOI: 10.1371/journal.pcbi.0020144. URL: https://doi.org/10.1371/journal.pcbi.0020144.

[25] M. Oizumi, L. Albantakis, and G. Tononi. "From the Phenomenology to the Mechanisms of Consciousness: Integrated Information Theory 3.0". In: *PLOS Computational Biology* (2014).

[26] O. Sporns and T. K. Pegors. "Information-Theoretical Aspects of Embodied Artificial Intelligence". In: *Embodied Artificial Intelligence: International Seminar, Dagstuhl Castle, Germany, July 7-11, 2003. Revised Papers*. Ed. by F. Iida, R. Pfeifer, L. Steels, and Y. Kuniyoshi. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 74–85.

[27] G. Tononi. "An information integration theory of consciousness". In: *BMC Neuroscience* (2004).

[28] G. Tononi. "Consciousness as Integrated Information: a Provisional Manifesto". In: *Biol. Bull.* (2008).

[29] G. Tononi and G. M. Edelman. "Consciousness and Complexity". In: *Science* (1999).

[30] G. Tononi, O. Sporns, and G. M. Edelman. "A measure for brain complexity: Relating functional segregation and integration in the nervous system". In: *Proc. Natl. Acad. Sci. USA* (1994).

[31] H. Touchette and S. Lloyd. "Information-theoretic approach to the study of control systems". In: *Physica A: Statistical Mechanics and its Applications* 331.1 (2004).

[32] M. Toussaint. "Probabilistic inference as a model of planned behavior". In: *Künstliche Intell.* 23.3 (2009), pp. 23–29.

[33] M. Toussaint, S. Harmeling, and A. Storkey. *Probabilistic inference for solving (PO)MDPs*. Tech. rep. 934. School of Informatics, University of Edinburgh, Dec. 2006.

[34] M. Toussaint, C. Laurent, and P. Pascal. "Hierarchical POMDP Controller Optimization by Likelihood Maximization". In: *arXiv e-prints* (2012).