# Using MAP-Elites to direct the evolution of desired neural characteristics

Douglas Kirkpatrick[1,2] and Arend Hintze[2,3]

[1]Michigan State University, Department of Computer Science and Engineering, East Lansing, 48824, United States of America
[2]Michigan State University, BEACON Center for the Study of Evolution in Action, East Lansing, 48824, United States of America
[3]Dalarna University, Department of MicroData Analytics, Dalarna, 79188, Sweden
kirkpa48@msu.edu

## Abstract

Artificial cognitive systems (e.g., artificial neural networks) have taken an ever more present role in the modern world, providing enhancements to everyday life in our cars, in our phones, and on the internet. In order to produce systems more capable of achieving their designated tasks, previous work has sought to direct the evolution of networks using a process referred to as $\mathcal{R}$-augmentation. This process selects for the maximisation of an information-theoretic measure of the agent's stored understanding of the environment, or its representation ($\mathcal{R}$) in addition to selecting for task performance. This method was shown to induce increased task performance in a shorter amount of evolutionary time compared to a standard genetic algorithm. Extensions of this work have looked at how $\mathcal{R}$-augmentation affects the distribution of representations across the neurons of the brain "tissue" or nodes of the network, referred to as smearedness ($\mathcal{S}$). Here we seek to improve upon the prior methods by moving beyond the simple maximization used in the original augmentation formula by using the MAP-Elites algorithm to identify intermediate target values to optimize towards. We also examine the feasibility of using MAP-Elites itself as an optimization method as opposed to the traditional selection methods used with $\mathcal{R}$-augmentation, to mixed success. These methods will allow us to shape how the network evolves, and produce better-performing artificial cognitive systems.

## Introduction

While the recent interest, funding, and computational breakthroughs have driven artificial intelligence to new heights in both capability and widespread use, there is still a lack of understanding on how to best shape the behavior, structure, and performance of artificial cognitive systems to reflect modern needs. In order to broadly adapt artificial cognitive systems to a wide range of tasks and problems, we should focus our efforts on generating cognitive systems that understand the tasks that they are presented with. For inspiration to how we may generate artificial cognitive systems that possess such understanding, we seek inspiration from beyond computer science. In the fields of philosophy, psychology, and cognitive science, for example, researchers have stressed the importance of mental representations. In these fields, representations are the internal imagery of objects that are not currently perceived by the sensory organs. According to cognitive scientists (and many philosophers, e.g., Phillips and Singer (1997) or Pinker (2013)) these internal models of the world are essential for intelligent decision-making, and form the very foundation of human intelligence (Collins and Gentner, 1987). While the importance of mental representations for Artificial Intelligence (AI) has been clear from the outset, it has been proven difficult to program mental representations into computational substrates, leading proponents at the forefront of AI research to search for "Intelligence without Representation" (Brooks, 1991), an approach that has so far failed to generate truly intelligent systems. Instead, modern AI research has focused Convolutional Neural Networks (CNNs), which appear to be powerful classifiers of complex contextual visual scenes (LeCun et al., 1998). However, many of the researchers that were spearheading the "Deep Learning" revolution now concede that those networks are vulnerable to so-called "adversarial perturbations" (Jo and Bengio, 2018), and are unlikely to represent a step towards artificial general intelligence, because they do not develop robust mental models.

Recent research has shown that instead of directly programming mental representations, it is possible to produce them by evolving artificial brains that control agents that act and behave in complex simulated environments (Edlund et al., 2011; Marstaller et al., 2013; Albantakis et al., 2014; Tehrani-Saleh et al., 2018; Olson et al., 2016; Tehrani-Saleh et al., 2019). Within these digital brains (Markov Brains, see Hintze et al. (2017)) mental representations are used to make decisions in conjunction (and sometimes even without) reference to sensory information. The ability to mathematically define representations in terms of information theory (Marstaller et al., 2013) has made it possible to monitor the evolution of representations quantitatively, and to measure what is being represented in the brain even where those representations are to be found in subsets of neurons. As it turns out, how localized or distributed representations are can be quantified as well. Hintze et al. (2018) introduced a measure of smearedness ($\mathcal{S}$) that quantifies how much representations are localized within a single neuron or small group of neu-

rons, or conversely how much representations are smeared out over large groups of neurons. In previous work we showed that how smeared-out or distributed representations are depends on the type of cognitive system (Kirkpatrick and Hintze, 2020) and additionally this smearedness can affect the robustness of the computation (Kirkpatrick and Hintze, 2019b).

Furthermore, previous work shows that rewarding representations (referred to as $\mathcal{R}$) explicitly (along with other aspects of fitness) in a Genetic Algorithm (GA) leads to significantly better performance because "augmenting evolution with $\mathcal{R}$" encourages the evolution of AIs with robust mental models (see Schossau et al. (2015) or Kirkpatrick and Hintze (2019a)). However, many open questions remain. Is there an optimal balance between rewarding "R" and rewarding fitness? Should representations be modular (different concepts represented in different areas of the brain) or should they be non-local, "smeared out" over the brain, as it were? Here we focus on the role and impact of smearedness and how it can be used to augment genetic search.

One might assume that, like $\mathcal{R}$-augmentation, increasing smearedness is also advantageous. We will show that this is not the case, and that trying to minimize it is equally useless. Viewed in the context of neuro-correlate maximization introduced by Schossau et al. (2015), the smearedness $\mathcal{S}$ will not help us to improve the performance of the GA. However, we will show that integrating the neuro-correlates $\mathcal{S}$ and $\mathcal{R}$ with MAP-Elites optimization (Mouret and Clune, 2015) produces a novel, useful method to investigate this issue.

MAP-Elites was originally designed to improve performance by maintaining a diverse population of solutions. MAP-Elites optimization does not allow all solutions in the population to compete against each other, but instead creates a grid of solutions. The grid is superimposed on a multidimensional parameter space, where each parameter describes set of solutions binned by their performance on that parameter. The best performance for different combinations of parameter values are stored, creating a "Map of Elites". For example, instead of just evolving a predator for hunting success, one could also measure predator size and weight. When evaluating a predator's performance, the size and weight of the predator are also determined and are used to determine a bin on the grid that it is categorized into. This predator is then not competing against the rest of the population, but only against a previous solution with similar size and weight defined by the grid. When applying the MAP-Elites method using $\mathcal{R}$ and $\mathcal{S}$ as the dimensions for the grid, the search of the GA can be improved. Here we find that, perhaps as expected, the optimal solutions have an intermediary $\mathcal{S}$ rather than a maximal or minimal $\mathcal{S}$. However, surprisingly, those best solutions do not maximize $\mathcal{R}$ but are found to have intermediary values of $\mathcal{R}$ as well. This shows, that while maximizing or minimizing $\mathcal{S}$ does not improve the search of a GA, $\mathcal{S}$ is still a useful neurocorrelate when applying MAP-Elites selection. Further, using $\mathcal{S}$ in conjunction with $\mathcal{R}$ in MAP-Elites outperforms $\mathcal{R}$-augmentation in some cases.

# Methods

## Active Categorical Perception Task

The Active Categorical Perception Task (ACP) is the preeminent task used for evolution of artificial agents (especially Markov Brains) for the assessment of mental representations and $\mathcal{R}$ (Marstaller et al., 2013; Schossau et al., 2015; Hintze et al., 2018). In this task, an agent equiped with a sensor made from four sensor neurons arranged in two groups of two (with a blind spot in between) must either catch or avoid blocks falling diagonally at constant rate, depending on the size of the block (see Figure 1). Blocks of varying width (here, width 2 or width 4) are dropped one at a time towards the agent on a toroidal surface of height 34 units and width 16 units, with the blocks moving by 1 unit to the left or to the right as they fall 1 unit down towards the agent. The agent must identify the size and direction of the falling block, and position itself to either be under (i.e., catch) the small blocks or not under (i.e. avoid) the large blocks. As the agent cannot see the whole block at one point in time due to the blind spot, it must integrate information from successive timesteps in order to decide which action to take. Given the sizes of the blocks, the movement direction, and 16 different starting positions relative to the agent, there are 64 different condi-

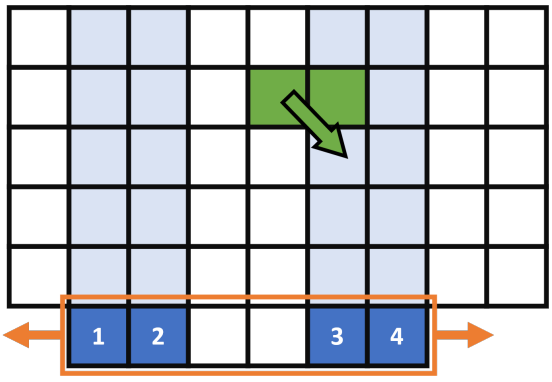


Figure 1: Visualization of a subsection of a single case of the Active Categorical Perception Task. The agent, visualized by the orange box, can move either to the left or right as indicated by the orange arrows. The four sensors, labelled 1,2,3, and 4, can each see directly upwards as indicated by the light blue shading. A small block with width 2 is falling towards the agent and moving to the left, as indicated by the green arrow. At the current point in time, sensor 3 is the only sensor that would return a signal, while on the next update, assuming that the agent did not move, both sensors 3 and 4 would return a signal.

tions that the agent is comprehensively evaluated for. Agents are assigned fitness based on the number of correct ($C$) and incorrect ($I$) catch/avoid decisions that they make, using an exponential fitness function (see Equation 1).

$$W = 1.10^{(C-I)} \tag{1}$$

## Numerical Comparison Task

The Numerical Comparison Task (NCT) is derived from real-world psychological (Merritt and Brannon, 2013) and biological (Nieder, 2018; Merritt et al., 2009) experiments with demonstrated utility in measuring $\mathcal{R}$ experimentally *in silico* (Kirkpatrick and Hintze, 2019b). In short, an agent receives one number and waits for a number of updates over which the agent is expected to remember the first number. After that time period the agent is given a second number, and again waits for the same number of updates, after which the agent must report which of the numbers was larger. *In silico*, the agent has a number of inputs, here 5, which are set to be either 0 or 1. The agent must keep track of how many of these inputs are provided as 1s (i.e., remember the number). There are a number of intermediary brain updates (here 3) after which the inputs are reset to have a different number of 1s. After a second intermediary period of brain updates (where length equal to the first period), the agent must provide an output with a value of 1 if the second input was larger than the first value, and a 0 otherwise. During the intermediary period, the agent may be provided with a constant signal (all 1s or 0s) or the input bits may be assigned randomly. For the experiments described herein, the agent was provided with random noise during the intermediary timesteps, with a bit being randomly assigned with probability $p = 0.5$. The agents are tested over all possible permutations of input variations (i.e., the number 2 could be input as 01100 or 11000, etc.) as well as permutations of relative number comparisons where unequal values are compared (i.e., 1 vs. 2, 2 vs. 1, 2 vs 5, etc.). We show an example of an agent being tested on a single permutation in Figure 2. The NCT is structured so that agents must receive information at two points in time, and then make a decision about them at a third time point, requiring information integration to occur between all three time points. Similarly to the ACP, agents are given fitness exponentially relative to the number of correct ($C$) and incorrect ($I$) decisions they make regarding which number is larger, again using the fitness function from Equation 1.

## Markov Brains

Markov Brains are a sparsely-connected evolvable neural network structure. These networks interact with their environment through sensory and motor neurons, which can receive input from and provide output to their environment, respectively. The number of input and output states in each Markov Brain is task-dependent. Here we use 4 input and 2 output for the Active Categorical Perception Task, and 5 input and 1 output for the Number Comparison task. In addition, there are a variable number of hidden neurons, here 8 for both tasks, which are only accessible internally and are recurrent to allow for information persistence and memory formation. These nodes are read-from or written-to by modular computational units that we call "gates". While a wide variety of gate types are available (e.g. single layer ANNs, Genetic Programming Functions, etc. - see Hintze et al. (2017) for examples), here we only use the deterministic and probabilistic logic gates to connect neurons.

## Mental Representations and Neuro-correlates

In order to quantify what the artificial cognitive system knows about its environment, we turn to information-theoretic methods. Previous work has demonstrated the utility of two different measures: $\mathcal{R}$, or representation, which measures the amount of information about the world that the system has integrated, and $\mathcal{S}$ or smearedness, which mea-

| Time | Sensors | Description |
|---|---|---|
| 1 | 1 0 1 1 0 | First Number |
| 2 | 1 0 0 1 0 | Noise |
| 3 | 1 1 1 0 1 | Noise |
| 4 | 0 0 1 0 0 | Noise |
| 5 | 1 1 0 1 1 | Second Number |
| 6 | 0 1 0 1 0 | Noise |
| 7 | 1 1 0 0 1 | Noise |
| 8 | 1 1 1 0 0 | Noise |
| 9 | 0 0 0 0 0 | Output Signal |

Figure 2: Visualization of a single case of the Numerical Comparison Task. The agent receives a series of inputs over a period of 9 brain updates. The input values are shown in the squares under "Sensors", where each square represents a unique sensor input, and correlate with brain updates listed under "Time". The description of each sensor input is likewise listed under "Description". The agent receives a first number, here 3 (note that 3 of the sensors have a 1 as input), and then has 3 intermediate updates where noise is fed to each sensor with probability $p = 0.5$. The agent then receives a second number, here 4 (note likewise that 4 of the inputs are 1), and then a second period of 3 intermediate updates where noise is fed to the sensors with the same probability $p$. After the second set of noise inputs, the agent receives an output signal of all '0' inputs, and is expected to output a 1 because the second number is larger than the first.

sures the distribution of that information across the hidden state space or memory of the system.

**Representations ($\mathcal{R}$)** Marstaller et al. (2013) identify an information-theoretic measure, called $\mathcal{R}$, that identifies the amount of mental representation that an agent has stored about its environment. The mental representation is defined to be the information shared between the agent's recurrent memory and the environmental states given the information currently coming from the agent's sensors. $\mathcal{R}$ can be visualized in the information-theoretic Venn diagram, Fig. 3. In practice, we use Equation 2 to calculate $\mathcal{R}$, which was introduced in previous work (Kirkpatrick and Hintze, 2020).

$$\mathcal{R} = H(S,B) + H(S,E) - H(S) - H(E,B,S) \quad (2)$$

In Eq. 2, $B$ refers to the hidden state or memory of the brain, $E$ refers to the salient features of the environment or task, and $S$ refers to the information coming from the sensors.

**Smearedness ($\mathcal{S}$)** We are not only interested in the amount of representations measured, but also investigate the structure of said representations by considering an additional information-theoretic measure, smearedness or $\mathcal{S}$ (Hintze et al., 2018) to quantify the distribution of mental representations across the hidden states of the artificial cognitive system. First, observe that the memory state $B$ can be deconstructed into individual neurons $B_i$, where $B = B_1 B_2 ... B_n$. Similarly, the environmental state $E$ can be deconstructed into individual concepts $E_j$ where $E = E_1 E_2 ... E_m$. $\mathcal{S}$ is calculated by first recording the 'atomic $\mathcal{R}$' or $\mathcal{R}_{ij}$, which is the information shared between *individual* brain states $B_i$ and *individual* concepts $E_j$ from the task, given the entire state of the sensors $S$. We then construct the matrix $M$ where $M_{ij} = \mathcal{R}_{ij} = H(B_i : E_j|S)$. We take a pairwise minimum over all concepts *i* and all nodes *j* and *k* to calculate $\mathcal{S}$ (see Equation 3).

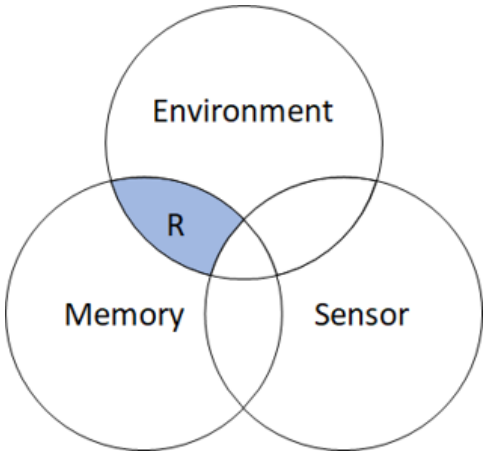


Figure 3: The information-theoretic Venn Diagram describing how $\mathcal{R}$ is calculated. $\mathcal{R}$ is defined as the information shared between the brain's memory and the salient features of the environment or task given the state of the sensors.

$$\mathcal{S} = \mathcal{S}_C = \sum_i \sum_{j>k} min(M_{ji}, M_{ki}) \quad (3)$$

This can alternatively be done pairwise over nodes and concepts, however due to the high correlation between the two different calculations (Kirkpatrick and Hintze, 2019b), we only use the first as $\mathcal{S}$ in the context of this paper.

## Augmented Fitness Functions

For both the ACP and NCT, the typical fitness function used (see Equation 1) takes into account the correct ($C$) and incorrect ($I$) decisions that the agent makes to determine the score. The traditional method for encouraging mental representations in the fitness function, referred to as $\mathcal{R}$-augmentation, involves multiplying the original function by a normalized term including $\mathcal{R}$ (Schossau et al., 2015). Schossau et al. find that the augmented function (Equation 4) results in the evolution of agents which have both higher levels of mental representation and better task performance.

$$W = 1.10^{(C-I)}(1 + \frac{\mathcal{R}}{\mathcal{R}_{max}}) \quad (4)$$

The augmentation function is modified with the addition of the $\mathcal{R}$ term so that the evolutionary process encourages higher values of $\mathcal{R}$ and consequently the formation of useful mental representations.

However, previous work (Hintze et al., 2018) shows that the structure of the mental representation may itself matter as much as the amount of representation present in the brain. Therefore, it seems plausible that we may also want to select for the structure of the representation. For example, we may choose to modify the augmented function to maximise the smearedness of the representations (see Equation 5).

$$W = 1.10^{(C-I)}(1 + \frac{\mathcal{S}}{\mathcal{S}_{max}}) \quad (5)$$

However, simple maximization may not work as well for the structure of the representations. Notably, results from Hintze et al. (2018) suggest that a *lower* value of smearedness may work better for network robustness. Thus, we may try a matching augmentation method that encourages the evolution of *minimal* smearedness. The original augmented fitness function is structured so that the term containing $\mathcal{R}$ is evaluated to a floating point value between 1 and 2, so we shall do the same for our minimisation augmentation, with the reversed importance of smearedness (i.e., higher smearedness results in a lesser evolutionary benefit and vice-versa). This formulation results in a new fitness function, Equation 6.

$$W = 1.10^{(C-I)}(2 - \frac{\mathcal{S}}{\mathcal{S}_{max}}) \quad (6)$$

Alternatively, an absolute minimum of smearedness may not be ideal either, as some smearedness and consequently information distribution across the brain is likely necessary to perform computations across (i.e., make decisions based on) multiple concepts from the environment. Following the same constraints as for the minimal smearedness modification, we in its stead propose a new augmented fitness formula, Equation 7, that evolves smearedness towards a hypothetical ideal target value, $T$.

$$W = 1.10^{(C-I)}(2 - \frac{|\mathcal{S} - T|}{max\{T, \mathcal{S}_{max} - T\}}) \quad (7)$$

We structure the equation so that the difference between the observed value of $\mathcal{S}$ and $T$ is minimized (i.e. the closer that $T$ and $\mathcal{S}$ are, the larger the increase to $W$). We also maintain the normalization between 1 and 2 for the multiplicative term, as in Equations 4, 5, and 6. Here, we used target values of 2.0, 4.0, 6.0, 8.0, and 10.0 based on previously observed values of smearedness.

### Genetic Algorithm

All experiments were implemented using the MABE software framework (Bohm et al., 2017). The circular genomes were restricted in size to be at least 2,000 sites and at most 20,000 sites long, where sites contained values in the range of 0 to 256. When we do not use MAP-Elites, we apply roulette-wheel selection to populations of 100 initially random agents with genomes of size 5,000. We apply mutations between generations using the default per-site point mutation rate of 0.005, insertion duplication rate of 0.00002 for chunks of size between 128 and 512, and a per-site deletion rate of 0.00002 for chunks with size between 128 and 512. After evolutionary optimization concluded, the line of descent (LOD) was recovered (Lenski et al., 2003), and subsequent analysis was conducted on agents from the LOD. We construct the LOD by maintaining a reference the parent of each organism as the agents are evolved, and subsequently finding the entire lineage of the best-performing agent at the end of evolution. For each task, the experiments were run for 20,000 generations, with 100 replicate experiments conducted for each task and augmentation method.

### MAP-Elites

The MAP-Elites algorithm (Mouret and Clune, 2015) works by maintaining a grid in multidimensional space (here, we use $\mathcal{R}$ and $\mathcal{S}$ as the axes of a 2-dimensional space), where individual squares along the grid contain the highest-performing individual on a different measure (here the task fitness from Equation 1, $W$). After starting with an empty grid, 100 randomly generated Markov Brains are evaluated and placed on the 2-dimensional grid according to their respective values of $\mathcal{R}$ and $\mathcal{S}$ using bins of fixed width for each dimension. The intersection of one bin from the $\mathcal{R}$ axis (e.g., where $\mathcal{R}$ is between 1.4 and 1.6) and one bin from the $\mathcal{S}$ axis (e.g., where $\mathcal{S}$ is between 6.3 and 7.2) constitute a cell for that grid location, where the corresponding task score $W$ is recorded. Afterwards, we take a randomly-selected sample of 10 individuals from all of the agents in the grid, and then mutate them and evaluate them on the given task to generate scores for $\mathcal{R}$, $\mathcal{S}$, and $W$. Using these values for each new agent, we find the corresponding bin locations on the grid location, and if the new agent has a higher $W$ than the agent in the cell, or the cell is empty, we place the new agent at that spot on the grid. We repeat this process for 200,000 updates, and run 100 replicate experiments for each task. We averaged the values in the resulting heatmaps of the 100 replicate experiments, so that we take a more accurate estimation of the ideal values of $\mathcal{R}$ and $\mathcal{S}$. Based on the observation of values of $\mathcal{R}$ in previous work (Marstaller et al., 2013; Schossau et al., 2015; Kirkpatrick and Hintze, 2019b) and in accordance with preliminary experiments (data not shown), we select a range of 0.0 to 2.0 for $\mathcal{R}$, with 10 equally distributed bins across those values to provide proper resolution without subdividing too far. As $\mathcal{S}$ has been shown to have a greater range of potential values (Hintze et al., 2018; Kirkpatrick and Hintze, 2019b, 2020), we use a similar selection methodology to set the bins for smearedness to be more numerous and across a slightly larger range - 20 bins equally distributed over the range from 0.0 to 18.0. Any values below the range specified for either dimension are grouped together in 1 bin, and any values above the range specified are also grouped together, in a separate bin.

## Results and Discussion

Schossau et al. (2015) show that $\mathcal{R}$-augmentation can improve the performance of a genetic algorithm. Without $\mathcal{R}$-augmentation, mutations that improve a network's internal model often do not convey an immediate fitness advantage. When using $\mathcal{R}$-augmentation, those mutations are not neutral to evolution but instead contribute to fitness. Consequently, $\mathcal{R}$-augmentation accelerates genetic search (Schossau et al., 2015). $\mathcal{R}$-augmentation assumes that a system with a high $\mathcal{R}$-value is in general better than a system with a lower $\mathcal{R}$-value. Is the same true for smearedness? Mutations that affect information smearing could be neutral and only later, in conjunction with other mutations, provide a fitness advantage. However, we do not know if smearedness should be maximized or minimized. Previous work (Hintze et al., 2018) finds that computational systems which have sparse connections tend to evolve low $\mathcal{S}$, while dense systems like fully connected neural networks evolve to have a high degree of smearedness. To explore if $\mathcal{S}$ should be maximized or minimized we compare both options for augmentation with a control experiment without augmentation,

and an experiment using conventional $\mathcal{R}$-augmentation. We find that, as expected, $\mathcal{R}$-augmentation improves the performance of the genetic algorithm compared to the control on both tasks (see Figure 4, where the blue line represents the unaugmented control and the black line represents augmented data). At the same time, trying to augment by maximizing (using Eq. 5) or minimizing (using Eq. 6) smearedness results in worse performance than either $\mathcal{R}$-augmentation or the unaugmented control (see Figure 4, where the red line represents $\mathcal{S}$-maximization and the green line represents $\mathcal{S}$-minimization).

This confirms our intuition that intermediary levels of smearedness might be optimal as opposed to the extremes. To determine which value for smearedness is optimal, we repeated the experiment, but used an augmentation function that optimizes for a specific value of smearedness instead of just maximizing or minimizing it (see Equation 7). Specifically, the following target values were used: 2.0, 4.0, 6.0, 8.0, and 10.0.

We find several target values for smearedness that when used for augmentation perform equal to or better than the unaugmented control (see Figure 5) at varying points in evolutionary time. The target values of 4.0, 6.0, and 10.0 applied to the ACP task result in higher task performance than control at end of evolution, while the target value of 6.0 causes higher task performance than control at around 3000 generations into the evolutionary process, and equivalent performance to the unaugmented control at the end of evolution. We find that the exact target values of $\mathcal{S}$ that maximize performance seems to be task-dependent, and targeted augmentation only outperforms $\mathcal{R}$-augmentation on the ACP task, but not the NCT task. While this confirms that intermediary values can be used for augmentation towards a target value, such an approach is extremely wasteful with respect to computational resources: the performance gain is relatively small while many extra runs are needed to narrow in on the optimal value. This could mean that using augmentation towards a target value of $\mathcal{S}$ is only viable if the optimal value of $\mathcal{S}$ is known beforehand. It might be the case that there is an optimal value for smearing that is general for all computational systems, but from observations in other contexts we found no evidence for a consistent value of $\mathcal{S}$ across computational systems (data not shown).

As a simple sweep of values for smearedness may or may not produce an optimal value for the targeted evolution, and would require excessive computational time to verify, we elect to use a more comprehensive method, MAP-Elites, to identify a hypothetical ideal value for targeted evolution. Previous work (Mouret and Clune, 2015) shows that the MAP-Elite selection method explores fitness landscapes more effectively and potentially finds better solutions when other phenotypic or genotypic properties beyond task performance are measured. In our case, we use $\mathcal{R}$ and smearedness as the dimensions for the MAP-Elite space. Based on previ-

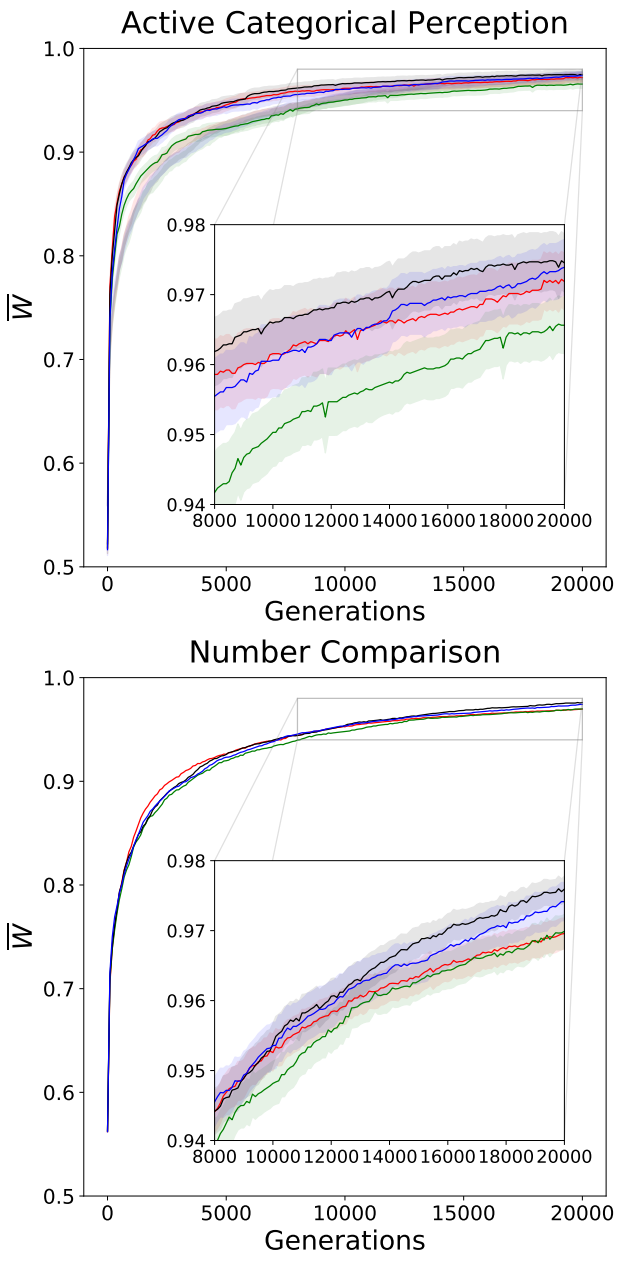


Figure 4: Average percent correct decisions over evolutionary time for the ACP and NCT tasks, where each line represents a different version of the fitness function used for that experiment. Each line is the average of 100 replicate runs using different random seeds. The blue line represents the original fitness function (Eq. 1). The black line represents $\mathcal{R}$-augmentation (Eq. 4). The red line represents $\mathcal{S}$-maximization (Eq. 5). The green line represents $\mathcal{S}$-minimization (Eq. 6). The shaded area of corresponding color around each line represents the standard error. The inset of the graph provides an expanded view of the indicated subset, to better distinguish the difference between conditions.

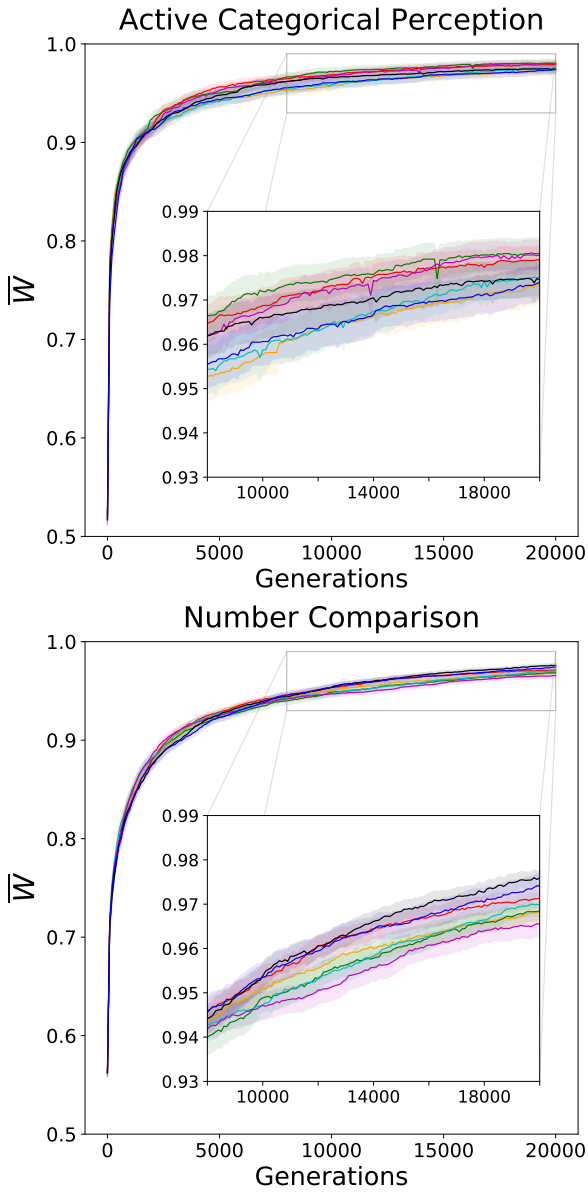


Figure 5: Average percent correct decisions over evolutionary time for the ACP and NCT tasks. Each line represents a different version of the fitness function used for that experiment and is the average of 100 replicate runs using different random seeds. As in Fig. 4, the black line represents $\mathcal{R}$-augmentation (Eq. 4), while the blue line represents the control. Each of the other differently colored lines represent $\mathcal{S}$-augmentation towards a target value (Eq. 7), for a range of different values. The correspondence of color to target values is: orange to 2.0, green to 4.0, red to 6.0, cyan to 8.0, and magenta to 10.0. The shaded area of corresponding color around each line represents the standard error. The inset of the graph provides an expanded view of the indicated subset, to exemplify the difference between conditions.

ous work (Kirkpatrick and Hintze, 2019b) and preliminary experiments, there are 10 bins for $\mathcal{R}$ in the range from 0.0 to 2.0, leading to a bin width of 0.2. Due to increased variance in the previously observed (Kirkpatrick and Hintze, 2019b; Hintze et al., 2018) values of $\mathcal{S}$, there are 20 bins for $\mathcal{S}$ over the range of 0.0 to 18.0, leading to a bin width of 0.9.

When evolving a populations of agents using MAP-Elites with $\mathcal{R}$ and smearedness as the dimensional variables, we find that as expected, optimal performing agents can be found for intermediary values of $\mathcal{S}$ (see Figure 6). However, surprisingly, those solutions that perform well with intermediary smearedness do not also have maximal values of $\mathcal{R}$. These results suggest that even though $\mathcal{R}$-augmentation works by maximizing $\mathcal{R}$, a maximized value of $\mathcal{R}$ might not even be the ideal goal for augmentation, and also that there is an interaction between the neuro-correlates $\mathcal{R}$ and $\mathcal{S}$.

We find that when using MAP-Elites, the best average performance over all replicate experiment heatmaps for a specific combination of $\mathcal{S}$ and $\mathcal{R}$ in the ACP is 99.66% correct decisions and in the NCT is 86.66% correct decisions. This result can be directly compared to the previous optimizations using a regular GA or $\mathcal{R}$-augmentation (see Figure 4). In both cases 2,000,000 solutions were evaluated in total, and the average performance for the non MAP-Elite cases from an unaugmented GA were 97.41% for the ACP, and 97.39% for the NCT. Further, $\mathcal{R}$-augmentation improves the outcome performance by 0.5% or 1% for the NCT and ACP, respectively. While the absolute differences are not necessarily significant on their own, $\mathcal{R}$-augmentation finds the same performance as unaugmented evolution much earlier. When evaluated using the Mann-Whitney U test, we find that the performance at end of evolution for the agents evolved on the ACP task using MAP-Elites is significantly higher than the performance at end of evolution for the agents evolved on the ACP task using $\mathcal{R}$-augmentation ($p < 0.0001$). This shows that optimization using MAP-Elites using $\mathcal{R}$ and $\mathcal{S}$ can outperform $\mathcal{R}$-augmentation. Although we do not see the same effect when using MAP-Elites for the NCT task, it stands to reason that a more optimal setup of MAP-Elites may produce a better result for that task also. Notably, the heatmap for NCT in Fig. 6 shows a large amount of unused space where high values of $\mathcal{S}$ (e.g., greater than 10.0) are not reached. This indicates that the range of values chosen may be too large, reducing the selective pressure on the agents. This discrepancy in MAP-Elites performances on both tasks related to the range of $\mathcal{S}$ may also be linked to the nature of the tasks, as the ACP task requires more information integration than the NCT task, and as such can tolerate a larger range of values for $\mathcal{S}$. The identification of the reason for the performance discrepancy between the two tasks, and more broadly the further refinement of the MAP-Elites method using neurocorrelates for generalized applications remains an open direction for future research.

## Conclusion

Using neurocorrelates to improve the ability of a genetic algorithm to search for optimal solutions is a promising direction for current and future research. As $\mathcal{R}$-augmentation showed, a simple joint fitness function already improves performance. Naturally, one might think that adding more complex neurocorrelates would improve the augmentation process further. However, we have shown that maximizing or minimizing for $\mathcal{S}$ is not a rewarding endeavor. Although it appears that using the augmentation towards an optimal target neurocorrelate value for either $\mathcal{S}$ or $\mathcal{R}$ could produce a better outcome than either evolution alone or even the original $\mathcal{R}$-augmentation method, the lack of *a priori* knowledge about optimal target values limits the applicability of this method. MAP-Elites serves as a potential method to overcome these issues both in the process of identifying ideal target values and also as an alternate evolutionary scheme whose performance is on par with $\mathcal{R}$-augmentation. Measuring $\mathcal{R}$ and $\mathcal{S}$ during evolution and using it to define the "Map of Elites" not only finds which values for $\mathcal{R}$ and $\mathcal{S}$ correlate with finding optimal solutions, the method itself outperforms a standard genetic algorithm as well as $\mathcal{R}$-augmentation. MAP-Elites may provide a better estimation of the target value without an exhaustive search, but some *a prior* knowledge is necessary to set the bounds of the grid.

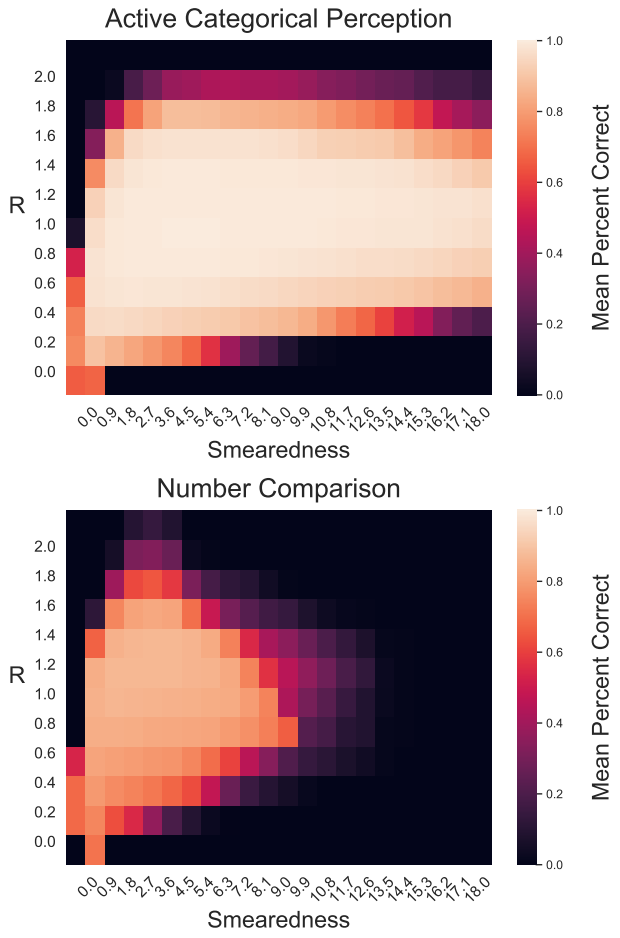


Figure 6: Heatmap of average MAP-Elite Results for the Active Categorical Perception Task and the Number Comparison Task. The top heatmap represents results from the ACP task, while the bottom heatmap represents results from the NCT task. Each rectangle represents the average value for a bin in MAP-Elites, for some subset of the range of $\mathcal{R}$ on the Y-axis and $\mathcal{S}$ on the X-axis. The value is the average percentage of correct decisions, with the highest average value occurring where $\mathcal{R}$ is between 0.8 and 1.0 for the ACP, and between 1.0 and 1.2 for the NCT task; and where $\mathcal{S}$ is between 3.6 and 4.5 for the ACP, and between 2.7 and 3.6 for the NCT task.

Obviously, the neurocorrelates $\mathcal{R}$ and $\mathcal{S}$ can only be used in systems that have recurrent hidden states, as only then can these values be properly measured. Successfully measuring these neurocorrelates depends on the ability of the experimenter to determine the salient features of the environment that should be reflected in mental representations. But given these limitations, using MAP-Elites in conjunction with these neurocorrelates is a promising approach. The future will tell to what degree other neurocorrelates could be used. An example of an alternative neurocorrelate is information integration (Albantakis et al., 2014) which also does not necessarily linearly correlate with the performance of an agent(Joshi et al., 2013). Another limitation is that here we only used two different simple computational tasks, active categorical perception and number comparison. While we see no reason that this method should not work for other tasks, it would be good for future research to use this method on different tasks to confirm that it performs well. Additionally, although the focus of this paper was on Markov Brains, given that previous work (Kirkpatrick and Hintze, 2020, 2019b) has shown that the information-theoretic tools used here work on other computational substrates as well, we can identify a number of open questions that seek to answer how we can best use this method to evolve other types of artificial cognitive systems (e.g., RNN, LSTM, or any other recurrent network architecture). To conclude, the methods presented here broadly allow for the shaping of representation quantity and structure, and are not limited to any one task or cognitive system.

## Acknowledgements

The authors would like to thank Chris Adami for his input and guidance on the paper.

This work was supported in part by Michigan State University through computational resources provided by the Institute for Cyber-Enabled Research. This material is based in part upon work supported by the National Science Foundation under Cooperative Agreement No. DBI-0939454.

# References


Albantakis, L., Hintze, A., Koch, C., Adami, C., and Tononi, G. (2014). Evolution of integrated causal structures in animats exposed to environments of increasing complexity. *PLoS computational biology*, 10(12):e1003966.

Bohm, C., CG, N., and Hintze, A. (2017). MABE (modular agent based evolver): A framework for digital evolution research. *Proceedings of the European Conference of Artificial Life*.

Brooks, R. A. (1991). Intelligence without representation. *Artificial intelligence*, 47(1-3):139–159.

Collins, A. and Gentner, D. (1987). How people construct mental models. *Cultural models in language and thought*, 243:243–265.

Edlund, J. A., Chaumont, N., Hintze, A., Koch, C., Tononi, G., and Adami, C. (2011). Integrated information increases with fitness in the evolution of animats. *PLoS Comput Biol*, 7(10):e1002236.

Hintze, A., Edlund, J. A., Olson, R. S., Knoester, D. B., Schossau, J., Albantakis, L., Tehrani-Saleh, A., Kvam, P., Sheneman, L., Goldsby, H., et al. (2017). Markov brains: A technical introduction. *arXiv preprint arXiv:1709.05601*.

Hintze, A., Kirkpatrick, D., and Adami, C. (2018). The structure of evolved representations across different substrates for artificial intelligence. In *Artificial Life Conference Proceedings*, pages 388–395. MIT Press.

Jo, J. and Bengio, Y. (2018). Measuring the tendency of CNNs to learn surface stastistical regularities. arXiv:1711.11561.

Joshi, N. J., Tononi, G., and Koch, C. (2013). The minimal complexity of adapting agents increases with fitness. *PLoS Comput Biol*, 9(7):e1003111.

Kirkpatrick, D. and Hintze, A. (2019a). Augmenting neuroevolutionary adaptation with representations does not incur a speed accuracy trade-off. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, pages 177–178.

Kirkpatrick, D. and Hintze, A. (2019b). The role of ambient noise in the evolution of robust mental representations in cognitive systems. In *to appear in the Proceedings of the Artificial Life Conference 2019*. Cambridge, MA: MIT Press.

Kirkpatrick, D. and Hintze, A. (2020). The evolution of representations in genetic programming trees. In *Genetic Programming Theory and Practice XVII*, pages 121–143. Springer.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Lenski, R. E., Ofria, C., Pennock, R. T., and Adami, C. (2003). The evolutionary origin of complex features. *Nature*, 423:139–144.

Marstaller, L., Hintze, A., and Adami, C. (2013). The evolution of representation in simple cognitive networks. *Neural computation*, 25(8):2079–2107.

Merritt, D. J. and Brannon, E. M. (2013). Nothing to it: Precursors to a zero concept in preschoolers. *Behavioural processes*, 93:91–97.

Merritt, D. J., Rugani, R., and Brannon, E. M. (2009). Empty sets as part of the numerical continuum: conceptual precursors to the zero concept in rhesus monkeys. *Journal of Experimental Psychology: General*, 138(2):258.

Mouret, J.-B. and Clune, J. (2015). Illuminating search spaces by mapping elites. *arXiv preprint arXiv:1504.04909*.

Nieder, A. (2018). Honey bees zero in on the empty set. *Science*, 360(6393):1069–1070.

Olson, R., Adami, C., and Tehrani-Saleh, A. (2016). Flies as ship captains? digital evolution unravels selective pressures to avoid collision in drosophila. In *Proceedings of the Artificial Life Conference 2016 13*, pages 554–561. MIT Press.

Phillips, W. and Singer, W. (1997). In search of common foundations for cortical computation. *Behav Brain Sci*, 20:657–683.

Pinker, S. (2013). *Learnability and Cognition, new edition: The Acquisition of Argument Structure*. MIT press.

Schossau, J., Adami, C., and Hintze, A. (2015). Information-theoretic neuro-correlates boost evolution of cognitive systems. *Entropy*, 18(1):6.

Tehrani-Saleh, A., LaBar, T., and Adami, C. (2018). Evolution leads to a diversity of motion-detection neuronal circuits. In *Artificial Life Conference Proceedings*, pages 625–632. MIT Press.

Tehrani-Saleh, A., McAuley, J. D., and Adami, C. (2019). Mechanism of perceived duration in artificial brains suggests new model of attentional entrainment. *bioRxiv*, page 870535.