Evolution of Embodied Agents on a Numerical Cognition Task: Sequential Counting

Eduardo J. Izquierdo

Rose-Hulman Institute of Technology
izquierd@rose-hulman.edu

Abstract

This paper investigates the capability of embodied agents to perform a sequential counting task. Drawing inspiration from honeybee studies, we present a minimal numerical cognition task wherein an agent navigates a 1D world marked with landmarks to locate a previously encountered food source. We evolved embodied artificial agents controlled by dynamical recurrent neural networks to be capable of associating a food reward with encountering a number of landmarks sequentially. To eliminate the possibility of the evolved agents relying on distance to locate the target landmark, we varied the positions of the landmarks across trials. Our experiments demonstrate that embodied agents equipped with relatively small neural networks can accurately enumerate and remember up to five landmarks when encountered sequentially. Counter to the intuitive notion that numerical cognition is a complex, higher cortical function, our findings support the idea that numerical discrimination can be achieved in relatively compact neural circuits.

Introduction

Numerical cognition, the ability to process and reason about quantities, offers a compelling window into the fundamental mechanisms of intelligence (Kadosh and Dowker, 2015; Cantlon et al., 2009). While traditionally viewed as a hallmark of complex brains, many species from diverse and distantly related animal groups, from bees and ants to monkeys and humans, have numerical sense (Nieder, 2021b). This widespread competence across the animal kingdom suggests that counting serves as a core building block for cognition. By examining how fundamentally different types of brains of animals and artificial systems represent and manipulate numerical information, we can gain profound insights into the essence of intelligence itself.

Scientists studying numerical cognition have employed a variety of techniques to assess counting abilities in both humans and animals (Nieder, 2020, 2021a). For honeybees, specifically, researchers have designed intricate experiments that tap into their impressive cognitive repertoire, and provide a fascinating glimpse into their sophisticated numerical processing abilities (Chittka and Geiger, 1995; Skorupski et al., 2017). One capacity that has been studied in detail is the honeybees' ability to count. In one study (Dacke and Srinivasan, 2008), researchers trained bees to associate a food reward with encountering a specific number of landmarks during flight. The distance to the reward was constantly changed, but the number of landmarks remained the same. This ensured the bees could not rely on distance to find the food. These experiments have revealed that bees can accurately count up to four objects when they encounter them in sequence.

In this paper, we set out to develop an embodied numerical competency task. Taking inspiration from the studies performed in honeybees (Dacke and Srinivasan, 2008), we focused specifically on sequential counting. To the best of our abilities, we developed a minimal version of the task that allows us to examine this cognitive ability. We deliberately eliminated all aspects of the agent and environment unrelated to counting, and we simplified all the relevant ones. Our goal with this task is to evolve agents that can be trained to remember the presence of a reward found in one of several possible landmarks encountered sequentially.

More broadly, one important contribution of this paper is to expand the repertoire of minimally cognitive behaviors to include numerical competency, a core cognitive capacity. In particular, the approach here is to study the simplest versions of behaviors that raise cognitively interesting issues using complete brain-body-environment models (Beer, 1996, 2021), which include visually guided behaviors, tasks requiring short-term memory, selective attention, social coordination and communication, decision-making, multi-functionality, context switching, and lifetime learning.

The task, agent, neural model, and evolutionary process that we employ are described in the next section. The results from our computational experiments are discussed in three parts. In Part I, we present the evolutionary results across different task difficulties and neural circuit sizes. In Part II, we examine the behavior of the most successful solutions. The robustness of the operation of the ensemble of successful circuits is then analyzed in some detail in Part III. The final section concludes with a discussion of the broader implications of our results and directions for future work.

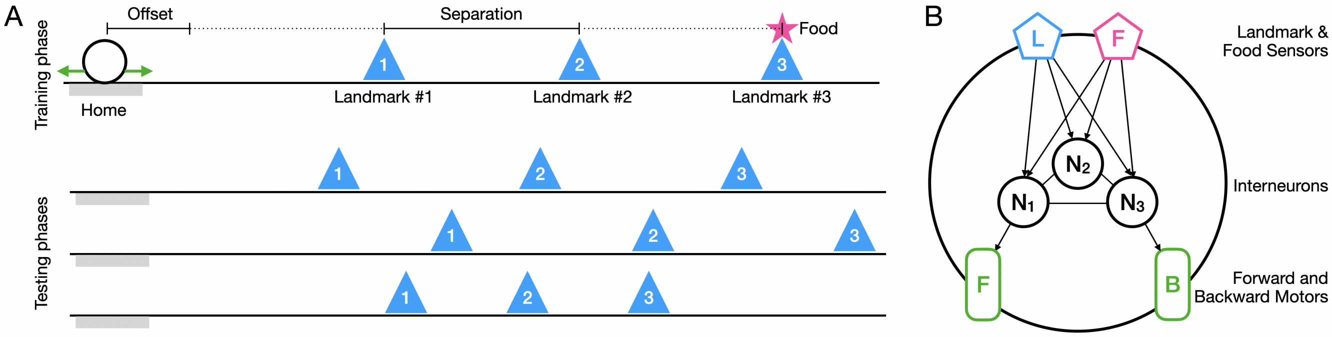

Figure 1: Task and agent setup. (A) Task takes place in 1D world with K landmarks and a food item, and an agent that can move forward and backwards in it. A trial encompasses first a training phase and then a testing phase. During the training phase, the agent is placed in the 'home' location (gray rectangle) and allowed to explore the space freely. During the testing phase, the food item is removed and the agent is relocated to the home location. The agent is required to find and remain near the landmark that had been associated with food during the training phase. The agent's proximity to that landmark is measured after an initial transient has passed. Landmarks are indistinguishable from each other. The absolute and relative position of landmarks is varied across trials, such that an agent can only succeed at this task if it can count the number of landmarks. (B) The agent can sense landmarks with one sensor (blue) and it can sense the food with another sensor (magenta). The agent has N neurons (black), fully interconnected, including self-connections (not shown). Both sensors have connections to all neurons. Two of the neurons drive a forward and backward motor, one each respectively.

Model

In this section, we first describe the original study we used for inspiration. We then describe our idealized task, environment, agent, neural network, and evolutionary training process in detail.

Original study overview The design for our minimal model of sequential counting is motivated by experiments performed by Dacke and Srinivasan (2008) on honeybees, as a follow-up to the first study showing this cognitive capacity (Chittka and Geiger, 1995). Bees (*Apis mellifera L.*) were trained to forage from a tunnel placed outdoors, which contained a series of prominent landmarks. For each experiment, up to 30 individually marked bees were trained to enter the tunnel and receive a food reward at one of the landmarks. The food was provided by placing a small container at the base of the rewarded landmark. The reward-bearing landmark was identical in appearance to the other landmarks, which carried no reward. Separate groups of bees were trained on tunnels containing a reward in either the first, second, third, fourth, or fifth landmark. Landmarks were placed at regular intervals. Bees were trained for a minimum of 3 and a maximum of 5 days. The training was considered complete when no improvement was observed between trials. After training, bees were tested individually in a tunnel with no reward. The time delay between training and testing is not discussed in the studies.

Sequential counting task The task takes place in a 1-dimensional world where an agent can move and encounter landmarks and food (Fig. 1A). A trial encompasses two phases: a *training phase* and a *testing phase*. Each phase lasts 300 units of time. During the training phase, an agent is placed in the 'home' location and allowed to explore the space, which includes K landmarks and a single food item that is located on any one of the landmarks. During the testing phase, the food item is removed and the agent is relocated to the home location, and tasked with finding the same landmark where food had been located originally.

The landmarks are indistinguishable from each other. Additionally, across trials the absolute and relative position of the landmarks is varied. The position of the landmarks is determined as follows:

$$\text{position}_i = \text{offset} + i \times \text{separation}$$

where position_i is the x-position of the i^{th} landmark, offset represents an initial distance from the starting position, and separation represents the distance between landmarks.

The design of the task is such that an agent can only succeed if it can identify the landmarks by their sequential ordering. In other words, the task requires that a successful agent be able to count the number of landmarks, remember which landmark contained food, and use this information to find the landmark again in the absence of food.

Agent An agent can sense landmarks and food and it can move forward and backward motor in its environment (Fig. 1B). The landmarks and food are two units of space wide. As in previous studies with agents perceiving in 1D environments (Izquierdo et al., 2022; Severino et al., 2023;

Merritt et al., 2023), we avoid discontinuities in the dynamics of the system by using the following sensor activation:

$$s_x(d) = \frac{1}{1 + e^{5(2d-1)}} \tag{1}$$

where d is the distance from the midpoint of the agent to the midpoint of the nearest object, s_x is the sensor activation value, and x represents the identity of the sensor: landmarks (s_L) or food (s_F). If the distance is greater than 2, the sensor activation is set to 0.

The behavior of the agent is controlled by a continuous-time recurrent neural network (Beer, 1995) with the following state equation:

$$\tau_i \dot{y}_i = -y_i + \sum_{j=1}^{N} w_{ji}\sigma(y_j + \theta_j) + l_i s_L + f_i s_F \tag{2}$$

where y_i is the state of each of N neurons, τ is the time-constant, w_{ji} is the connection weight from the j^{th} neuron to the i^{th}, θ is the bias term, $\sigma(x) = 1/(1+e^{-x})$ is the standard logistic activation function, l_i is the connection weight from the landmark sensor s_L to the i^{th} neuron, and f_i is the connection weight from the food sensor s_F to the i^{th} neuron. The output of a neuron is $o_i = \sigma(y_i + \theta_i)$. The network is fully inter-connected, including self-connections, and each sensor has a single weighted connection to every neuron.

Evolutionary process The neural parameters of the controller were evolved using a real-valued genetic algorithm. Each genome encodes the parameters for a neural controller. The following neural parameters, with corresponding ranges, are evolved: time-constants $\tau \in [1, 15]$, biases $\theta \in [-16, 16]$, and all connection weights (from sensors to neurons, l_i and f_i, and between neurons, $w_{ij}) \in [-16, 16]$. We used a generational algorithm with rank-based selection and a population size of 96 genotypes. Successive generations are formed by first applying random Gaussian mutations to each parent genome with a mutation variance of 0.05 (see Beer (1996) for details). In addition, uniform crossover is applied with 50% probability.

Fitness function A fitness evaluation for an agent includes several trials. Each trial consists of a training phase and a testing phase. The performance of an agent on a trial is evaluated using the proximity of the agent to the target landmark over the last 150 units of time of the training phase. The target landmark on any trial is determined by the location of the food during the training phase. For example, if the food was placed on the second landmark during the training phase, the second landmark would be the target landmark, regardless of its physical location. The fitness of an agent is proportional to the performance across all the possible combinations of trials:

$$f = \sum_{k=1}^{K} \sum^{O} \sum^{S} \int_{t=150}^{300} |x_t - p| \tag{3}$$

where K is the number of landmarks and k represents trials where the food is placed in each of the different landmarks during the training phase; O and S are the number of different offset and $\mathrm{separation}$ values used to vary the position of the landmarks during the testing phase, respectively; t from 150 to 300 represents time during the second half of the training phase; and x_t represents the position of the agent at time t and p represents the position of the target landmark during the training phase. Finally, an agent's fitness is normalized to run between 0 and 1.

Shaping protocol In preliminary experiments, we learned that evolving agents to succeed at this task was not trivial. To increase our chances of success, we implemented a staging protocol, which allowed us to increase the complexity of the task gradually. The shaping protocol includes four stages. In the first stage, the positions of the landmarks are fixed to a specific location for both the training phase and the testing phase across all trials ($\mathrm{offset} = 15$ and $\mathrm{separation} = 15$). The placement of the food during the training phase varies across trials, with all K possibilities evaluated. The stage changes as soon as the fitness of the best individual in the population surpasses a threshold of 0.99. In the second stage, we introduce small changes in landmark placement during the training phase: $\mathrm{offset} = \{14, 15, 16\}$ and $\mathrm{separation} = \{14, 15, 16\}$. The goal is to gradually encourage agents to ignore the position of the landmarks and to pay attention to their sequential order instead. We continue this gradual increase in complexity in the third stage, by increasing the range of changes in landmark placement during the training phase: $\mathrm{offset} = \{13, 15, 17\}$ and $\mathrm{separation} = \{13, 15, 17\}$. In the final stage, to further introduce resilience into the learning and memory component of the task, we introduce a time delay between the training phase and the testing phase of $\Delta = \{0, 5, 10\}$ units of time. The same 0.99 threshold was used for all stages.

Part I: Evolution

Our first goal was to determine whether we could evolve embodied agents controlled by dynamical recurrent neural networks in a numerical competence task. In this section, we report on the evolutionary performance across varying levels of difficulty (Fig. 2). Specifically, we performed experiments using three, four, and five landmarks. For each condition, we performed evolutionary runs with multiple circuit sizes. We used circuits with 3-5 neurons for the three-landmark task; 3-6 neurons for the four-landmark task; and 3-7 neurons for the five-landmark task. For every combination of circuit size and task difficulty, we performed one hundred evolutionary runs from different initial seeds (1,200 in total). Each run was performed for 10,000 generations.

The evolutionary algorithm found successful circuit configurations for all the task difficulty levels attempted. For the simpler task conditions, where agents have to evolve to

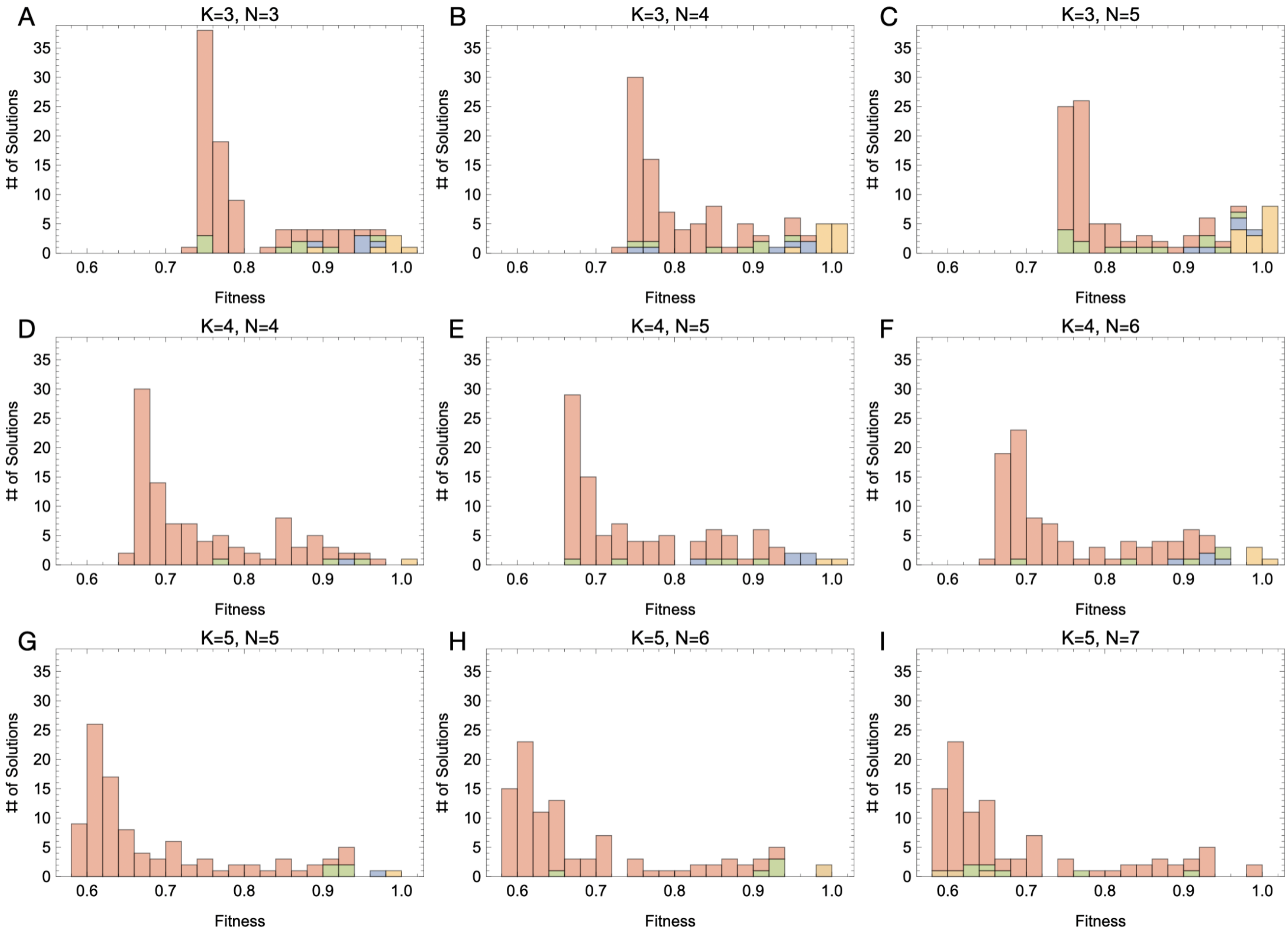

Figure 2: Evolutionary performance statistics. Histograms of the final performance achieved by the best individual in each evolutionary run. Each panel represents a specific condition. Rows represent task difficulty (number of landmarks, K): $K = 3$ (top row), $K = 4$ (middle row), $K = 5$ (bottom row). Columns represent number of neurons in the circuit (N), see label on top of each histogram. For each condition, results from 100 independent evolutionary runs are shown. Specifically, the performance of the best individual as evaluated on the final stage settings is shown for each evolutionary run. Colors indicate the highest stage reached by each evolutionary run: red, first stage; green, second stage; blue, third stage; yellow, fourth and final stage (most successful).

count up to three landmarks ($K = 3$), solutions evolved most readily. We observed successful solutions across all circuit size conditions tested ($N = 3, 4, 5$). Notably, success increased with circuit size. For three-neuron circuits, 6 of the evolutionary runs reached the final stage (Fig. 2A) and 3 of those 6 achieved a fitness on the final stage greater than 0.99. For four-neuron circuits, 11 runs reached the final stage Fig. 2B), and 9 of those achieved a fitness greater than 0.99. For the five-neuron circuits, 15 reached the final stage Fig. 2C), and 10 had a fitness greater than 0.99. Henceforth, we consider a solution successful if it meets two criteria: (a) its evolutionary run reached the final stage; and (b) it has a final fitness greater than 0.99.

As task difficulty increased with the number of landmarks, it became harder for evolution to discover successful solutions. Despite the difficulty, we found successful solutions for all tasks. For the four-landmark task ($K = 4$), we found: no successful solutions using three-neuron circuits (not shown); one successful four-neuron solution (Fig. 2D); two successful five-neuron solutions (Fig. 2E); and four six-neuron solutions (Fig. 2F). For the most difficult condition attempted in this study, the five-landmark task ($K = 5$), we found: no successful solutions using three- or four-neuron circuits (not shown); one successful five-neuron solution (Fig. 2G); two successful six-neuron solutions (Fig. 2H); and no successful seven-neuron solutions (Fig. 2I).

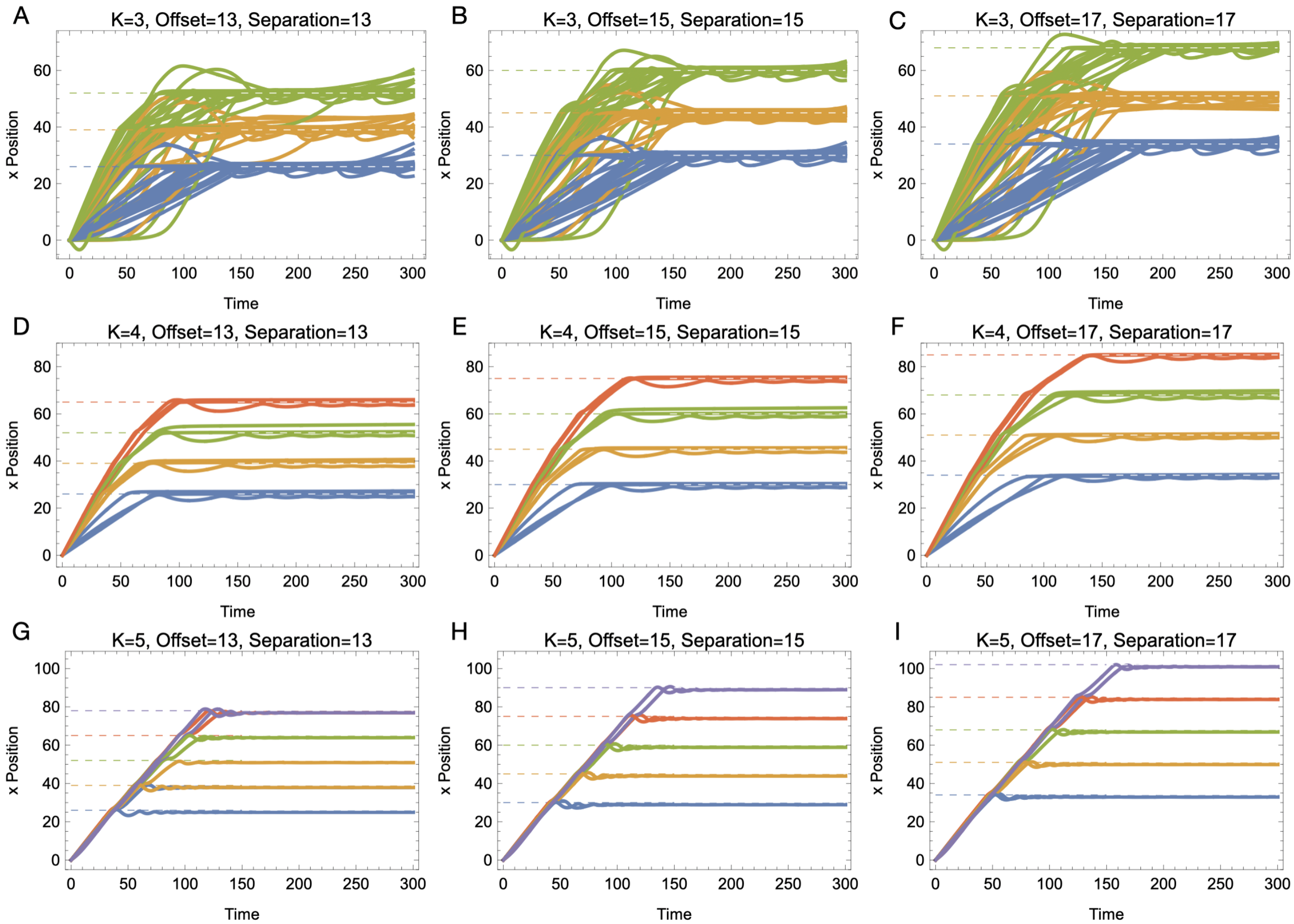

Figure 3: Behavior of successful solutions. Each panel depicts the physical trajectory of successful solutions (final fitness greater than 0.99, regardless of circuit size) in space (y-axis) and over time (x-axis) during the testing phase (while the agent is trying to find the landmark where food was present during the training phase). The dashed lines depict the location of the landmarks. For every panel, a solution is shown as many times as landmarks in the environment, and colored according to the landmark where food was present for that trial during the training phase. Rows depict the varying task difficulties: three-landmark (top), four-landmarks (middle), and five-landmarks (bottom). Columns depict different landmark placement conditions (variations of their absolute placement and relative separation).

What other general trends are observed in the evolution of agents for numerical competence? Across all conditions, most evolutionary runs became stuck in Stage 1. The most likely reason for this was the strict criteria that we set for the transition between stages: the best individual in the population had to surpass a threshold of 0.99. Although our specific approach led to success (after preliminary variations that did not), the threshold (and more generally the staging protocol) is an important area that would benefit from in-depth study. Our results also suggest that increasing the size of the circuit does not always provide a significant improvement. We suspect that allowing neural circuits to be fully connected could be partly to blame for this result. So varying the connectivity is also an important area of future work.

Part II: Behavior

What are the overall tendencies observed in the behavior of successful agents that can count? As a first step in the analysis of this new task, we examined the behaviors of successful solutions across task difficulties (Fig. 3). In the figure, the rows depict the varying task difficulties: three-landmark (top), four-landmarks (middle), and five-landmarks (bottom). Each panel contains the physical trajectory of all the successful solutions, regardless of circuit size, in space and over time. Trajectories are shown for the testing phase only, when agents are rewarded for finding the target landmark in the absence of a food reward. Traces are color-coded according to the landmark where food was present for that trial during the training phase. The dashed lines depict the loca-

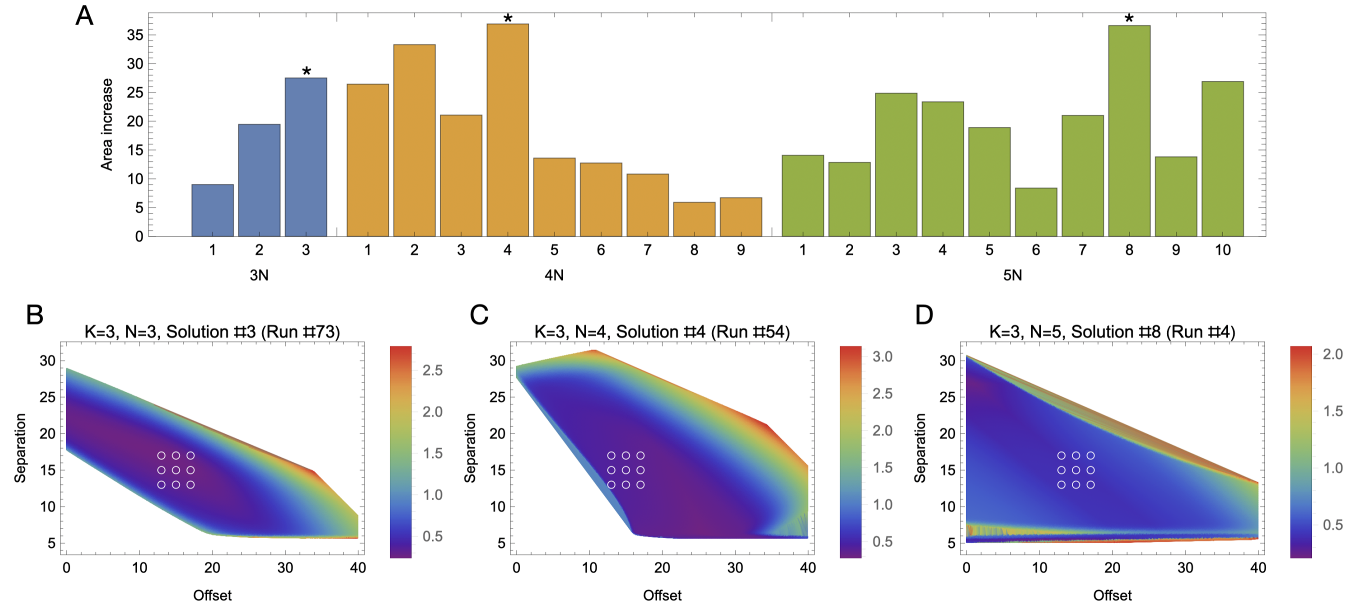

Figure 4: Generalization of successful solutions for the three-landmark counting problem. [A] Each agent was tested across a wider range of landmark placements and their ability to perform counting correctly was measured in relation to the original landmark placements they were trained with. The area increase in the space of possible offsets and separations is shown for each solution in the top as a bar chart. [B-D] The performance map used to determine the generalization is shown for the most successful solutions across each different neural circuit size. The white disks represent the 9 conditions that were used for training during evolution. The area shown in color represents additional offset and separation configurations that also led to successful behavior. The areas in white represent configurations where the behavior of the circuit breaks down.

tion of the previously reward-laden landmark using the same color coding. The columns depict some of the different landmark placement conditions that the agents experience during their evolutionary training. From the behavioral traces, it is evident that successful agents, starting from the same location at the beginning of the testing phase, move forward and identify the correct landmark, regardless of the relative positioning of the landmarks in the environment.

We highlight three key observations from the behavioral results. First, most of the circuits move forward until they encounter the target landmark and then stay either directly on the landmark or near it. A few circuits oscillate around the landmark, moving back and forth around it. A few circuits move some distance past the landmark first, and then re-center on it. Second, most of the circuits remain indefinitely on the landmark after some time has passed. A few circuits find the landmark and stay on it for most of the evaluation period, but towards the end of that period, they begin to move away from the landmark. Third, the behavior of any one solution is qualitatively similar across different placement variations. In other words, the way in which each agent approaches and finds the target landmark is consistent across different offset and separation conditions. Altogether, these results suggest these agents have successfully evolved to sequentially count.

Part III: Generalization

How robust are these counting agents to changes in the placement of the landmarks? Agents were evolved to cope with relatively small changes in the offset and separation of the placement of the landmarks. To test for generalization, we examined the ability of successful agents to cope with a much wider range of landmark placements. Specifically, we examined each agent while systematically varying the offset and $\mathrm{separation}$ each in the range $[0, 40]$ in steps of 0.1. This was repeated across each possible landmark training condition (food on the first, second, or third landmark during the training phase). This was also repeated across the range of different time delays between training and testing phases, $\Delta = \{0, 5, 10\}$. For each trial, we recorded the average distance to the target landmark. Altogether, the analysis for each agent comprised a total of 1,447,209 additional trials. We deemed an agent successful at a certain offset and separation configuration if the agent was within a distance of 5 units of space from the target landmark across all time delays and all possible landmark ordering. This provided us with an area of success in the 2-dimensional map of possible offsets and separations. We then calculated how much this area increased in relation to the baseline area that was used for training during evolution (offset and separation between [13,17]). We focused our generalization analysis on the 22 successful solutions trained on the three-landmark task. In Fig. 4, the area increase is shown for each of those agents. We also show example maps for the best three-neuron (Fig. 4B), four-neuron (Fig. 4C), and five-neuron circuit (Fig. 4D).

We highlight three key observations from the generalization analysis. First, all successful solutions generalized the sequential counting ability over a range of placements several times larger than what was provided during evolutionary training. The agent with the lowest score still generalized over an area 5.9 times larger than the area used during evolutionary training. Second, there are significant differences in generalization between different circuits, ranging between 5.9 and 36.9 times the original area. Third, the number of neurons in the circuit does not determine the ability of the circuit to generalize. For example, there are three-neuron circuits that generalize better than five-neuron circuits. Altogether, these results suggest: (a) that the ensemble of successful circuits is highly robust to changes in landmark placement, and (b) that there are potentially different counting strategies and neural implementations employed by different agents in the ensemble.

Related Work

The investigation into the sequential counting abilities of embodied agents using computational neuroethology modeling approaches remains relatively unexplored. Existing computational models of counting have predominantly focused on disembodied agents tasked with counting objects within static images (Fang et al., 2018; Sabathiel et al., 2020; Noda et al., 2024). Although some studies have incorporated agents capable of movement, including robotic entities (Pecyna et al., 2022), the counting process has remained primarily detached from the agent's interaction with its environment. An early study (Saggie-Wexler et al., 2006) explored an embodied agent capable of tracking time; however, the core counting aspect of the task also remained disembodied from the agent's interaction with the environment.

The most closely related studies involve computational models inspired by experiments also conducted in honeybees Howard et al. (2018), where they have to perform a "greater than" dual choice task on two stimulus images that show varying numbers of geometric shapes (circles, squares, diamonds). In their task, the input is a sequence of images. In Vasas and Chittka (2019), the authors design a four-neuron circuit to estimate numerosity in the image, with one neuron acting as an 'item counter'. In a follow-up study, Rapp et al. (2020) designed a synaptic plasticity rule and demonstrated that a single spiking neuron with this rule could perform a similar numerical computation. In both studies (Vasas and Chittka, 2019; Rapp et al., 2020), the neural mechanisms were crafted by hand to execute a particular counting strategy. The purpose of their computational models was to validate a specific hypothesis, in this case, a strategy and neural implementation for counting. In contrast, our endeavor in this work is to establish a framework for exploring the space of possible hypotheses regarding how counting could be realized in neural circuits when embodied and situated in their environments. Another important difference with previous work is that the artificial agents in our study, akin to the bees in their experimental environment (Chittka and Geiger, 1995; Dacke and Srinivasan, 2008), are trained during their lifetime. That is, unlike previous work, our study includes the learning and memory component of the sequential counting experiments which is fundamental to studies in bees and other living organisms.

Discussion

In this study, we investigated the ability of dynamical recurrent neural networks to control embodied agents in a numerical competence task. We developed a minimal version of a task motivated by sequential counting experiments in honeybees. We employed an evolutionary approach to explore the space of solutions across varying levels of task difficulty. We found an ensemble of successful solutions across all task difficulty levels using neural circuits with a small number of neurons. As a first step in the analysis, we visualized the behavior of all successful solutions and examined their ability to generalize the counting behavior for the three-landmark task. In this section, we discuss the main contributions of this work, the insights gained, the limitations of our study, and the directions for future work.

There are three main contributions of this work. First, we developed a minimal setup for a numerical competence cognitive task: sequential counting. This task is motivated by honeybees studies. More broadly it allows us to expand the repertoire of minimally cognitive behaviors. Second, we demonstrated that we can use a staged evolutionary process to train agents to succeed with up to 5 landmarks with just 5 neurons. Our findings provide evidence that embodied agents endowed with relatively small neural circuits are capable of counting. Third, we demonstrate that solutions are robust to variations in the absolute position of landmarks, their separation, and time delays between training and testing. Our findings suggest these agents can navigate by maintaining a running count of prominent landmarks that are passed en route.

One limitation of the work described in this paper is that the widths of the landmarks were not varied over trials. As we begin to understand how these agents perform the numerical cognitive task, varying the width of the landmarks will allow us to distinguish between agents integrating visual stimulus over time as a form of counting versus agents that are enumerating discretely. The variation of the landmark width can be performed during the analysis or during the evolutionary training. More broadly, one challenge of the current work was the difficulty of evolving agents that could count. As the number of possible landmarks increased, it was harder to obtain successful solutions. We believe this is part of a broader challenge for the neuroevolutionary approach. Although it is not the focus of this work to overcome this challenge, we believe this sequential counting provides an ideal task in which to systematically study neuroevolutionary approaches. In particular, this embodied numerical competence provides an ideal framework for testing claims about novel neural architectures and evolutionary search processes.

Finally, future work will focus on analyzing successful counting agents using dynamical systems theory and information theory. Several experimental and theoretical studies have suggested that numerical discrimination can be implemented in relatively simple circuits. We would like to answer the question of how these agents solve the counting task. The answer to this question is of central neurobiological interest. Of particular interest is to be able to explore the space of possibilities for how this can be accomplished in neural systems. The modeling approach employed in this study allows us to consider potentially different strategies and different neural implementations. Just as importantly, we would like to focus on analyzing not a single best counting agent, but an ensemble of successful ones. By examining how different neural circuits represent and manipulate numerical information, we hope to gain insights into this fundamental cognitive capacity. In the case of honeybees specifically, the hypothesis is that this cognitive capacity requires the ability to maintain a running tally of the number of events, incrementing the tally by one each time an event occurs. This is a hypothesis that we can test across the wide variety of solutions generated in the search process. Most importantly, by analyzing an ensemble of solutions, we may be able to suggest alternative hypotheses as well.

Acknowledgements

We thank Randall D. Beer for the discussions that led to the task studied in this paper. We thank members of the Computational Neuroethology research lab and anonymous reviewers for their feedback. We acknowledge Indiana University for providing supercomputing resources. This study was supported by NSF grant number 2345446.

References

Beer, R. D. (1995). A dynamical systems perspective on agent-environment interaction. *Artificial Intelligence*, 72(1-2):173–215.

Beer, R. D. (1996). Toward the evolution of dynamical neural networks for minimally cognitive behavior. In *From Animals to Animats 4: Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior*. The MIT Press.

Beer, R. D. (2021). Some historical context for minimal cognition. *Adaptive Behavior*, 29(1):89–92.

Cantlon, J. F., Platt, M. L., and Brannon, E. M. (2009). Beyond the number domain. *Trends in Cognitive Sciences*, 13(2):83–91.

Chittka, L. and Geiger, K. (1995). Can honey bees count landmarks? *Animal Behaviour*, 49(1):159–164.

Dacke, M. and Srinivasan, M. V. (2008). Evidence for counting in insects. *Animal Cognition*, 11(4):683–689.

Fang, M., Zhou, Z., Chen, S., and McClelland, J. (2018). Can a recurrent neural network learn to count things? In *Proceedings of the Annual Meeting of the Cognitive Science Society*.

Howard, S. R., Avarguès-Weber, A., Garcia, J. E., Greentree, A. D., and Dyer, A. G. (2018). Numerical ordering of zero in honey bees. *Science*, 360(6393):1124–1126.

Izquierdo, E. J., Severino, G. J., and Merritt, H. (2022). Perpetual crossers without sensory delay: revisiting the perceptual crossing simulation studies. Artificial Life Conference Proceedings, page 27.

Kadosh, R. C. and Dowker, A. (2015). *The Oxford Handbook of Numerical Cognition*. Oxford University Press.

Merritt, H., Severino, G. J., and Izquierdo, E. J. (2023). The dynamics of social interaction among evolved model agents. *Artificial Life*, 30(2):216–239.

Nieder, A. (2020). The adaptive value of numerical competence. *Trends in Ecology & Evolution*, 35(7):605–617.

Nieder, A. (2021a). The evolutionary history of brains for numbers. *Trends in Cognitive Sciences*, 25(7):608–621.

Nieder, A. (2021b). Neuroethology of number sense across the animal kingdom. *Journal of Experimental Biology*, 224(6):jeb218289.

Noda, K., Soda, T., and Yamashita, Y. (2024). Emergence of number sense through the integration of multimodal information: developmental learning insights from neural network models. *Frontiers in Neuroscience*, 18.

Pecyna, L., Cangelosi, A., and Nuovo, A. D. (2022). A robot that counts like a child: a developmental model of counting and pointing. *Psychological Research*, 86(8):2495–2511.

Rapp, H., Nawrot, M. P., and Stern, M. (2020). Numerical cognition based on precise counting with a single spiking neuron. *iScience*, 23(2):100852.

Sabathiel, S., McClelland, J., and Solstad, T. (2020). A computational model of learning to count in a multimodal, interactive environment. In *Proceedings of the Annual Meeting of the Cognitive Science Society*.

Saggie-Wexler, K., Keinan, A., and Ruppin, E. (2006). Neural processing of counting in evolved spiking and mcculloch-pitts agents. *Artificial life*, 12(1):1–16.

Severino, G. J., Merritt, H., and Izquierdo, E. J. (2023). Between you and me: A systematic analysis of mutual social interaction in perceptual crossing agents. In *ALIFE 2022: Proceedings of the Artificial Life Conference*, page 24.

Skorupski, P., MaBouDi, H., Galpayage-Dona, H. S., and Chittka, L. (2017). Counting insects. *Philos Trans R Soc Lond B Biol Sci*, 373:20160513.

Vasas, V. and Chittka, L. (2019). Insect-inspired sequential inspection strategy enables an artificial network of four neurons to estimate numerosity. *iScience*, 11:85–92.