Children successfully infer algorithmic properties based on only partial descriptions

Emily M. Sanford (esanford@berkeley.edu)[1], **Sarah Sommer**[1] **& Steve Piantadosi**[1]
[1]Department of Psychology, University of California, Berkeley

Abstract

Much of early childhood learning involves learning how to correctly carry out a series of steps—an algorithm—to reach a goal. It is not known whether the general ability to make inferences about novel algorithms develops in childhood, alongside the ability to execute new algorithms, or if the inferential capacity develops later in life. We tested 4- to 8-year-old children using a set of storybooks depicting animals playing procedural games, each corresponding to an algorithm with a definite set of possible ending states. Critically, the stories depicted only the first few steps of each procedure. Then, participants were asked questions about the properties of those algorithms. We found that children successfully made multiple types of inferences about novel algorithms. Even the youngest children were already sensitive to algorithmic properties. These results provide the first evidence that children understand that algorithms are more than just a series of steps.

Keywords: cognitive development; algorithmic reasoning; mathematics

Introduction

An algorithm is a well-defined sequence of steps that converts an input to an output. Algorithms are ubiquitous in everyday life—whether greeting a friend, cooking a recipe, or counting the number of items in a shopping cart, people execute hundreds of programs every day. Adults are competent algorithmic reasoners, and can both invent algorithms to solve novel problems and deduce the consequences of those algorithms (Khemlani et al., 2014).

Children acquire a vast amount of procedural knowledge over the course of development (Rule et al., 2020). In children, previous research on algorithmic cognition has been largely piece-meal, focusing on how they learn to carry out specific, usually mathematical, procedures; for example, counting (Piantadosi et al., 2012), addition (Rittle-Johnson et al., 2001), patterning (Wijns et al., 2019), seriation (Kingma, 1987), sorting (Yang et al., 2025), and geometry (Amalric et al., 2017).

In order to master an algorithm, a child must learn the sequence of steps it involves and be able to flexibly apply its steps to different inputs. However, simply being able to carry out an algorithm does not guarantee that one understands what the algorithm does, under what conditions the algorithm might fail, or whether another procedure would accomplish the same task more efficiently. While adults are sensitive to a broad range of algorithmic metrics such as efficiency and robustness to noise (Rule et al., 2020), it is not known whether this sensitivity develops in childhood alongside the development of the capacity to carry out algorithms, or whether the ability to make inferences about algorithms develops only after procedural competency is well established. Across the literature, the proposed order of acquisition of conceptual knowledge and procedural mastery vary widely. In counting, procedural mastery appears to develop prior to conceptual understanding (Wynn, 1989); in the development of integer computations, conceptual knowledge facilitates procedural development (Byrnes, 1992); and in decimal fractions, procedural abilities and conceptual understanding develop iteratively, supporting one another in turn (Rittle-Johnson et al., 2001).

Here, we presented young children (4 to 8 years old) with partial descriptions of novel algorithms, disguised as games played by animals in a storybook format. Critically, children never saw the algorithm carried out in its entirety. Then, we asked them to make inferences about the algorithms—what end state would result from a given starting state, which of two starting states would require the fewest number of steps to carry out, and what would happen if the procedure were applied to a different dimension. If children can make successful inferences about algorithms based only on partial descriptions, this would indicate that sensitivity to algorithmic properties is present early in life, and that familiarity with and mastery of an algorithm is not required in order for one to make inferences about it.

Methods

Participants

A total of N = 101 children participated in the experiment, with 20 4-year-olds (4;3-4;11, M = 4;8), 21 5-year-olds (5;1-5;11, M = 5;6), 24 6-year-olds (6;0-6;10, M = 6;5), 21 7-year-olds (7;0- 7;11, M = 7;6), and 14 8-year-olds (8;0-8;11, M = 8;6); one parent declined to provide their child's age. An additional 24 children were excluded from the experiment (technical difficulties such as camera failures, N = 5; experimenter error, N = 1; ineligibility, N = 1; failure to pass any training trials, N = 2; voluntarily withdrew, N = 14). The experiment was approved by the Institutional Review Board at the authors' university.

Materials

The stimuli consisted of stories presented in a slideshow on a computer screen. There were three different stories, each of which described an animal playing a procedural game (see

In G. Boleda, I. Dautriche, M. Maldonado, C. Saldana and T. Torrent (Eds.), *Proceedings of the 48th Annual Conference of the Cognitive Science Society*

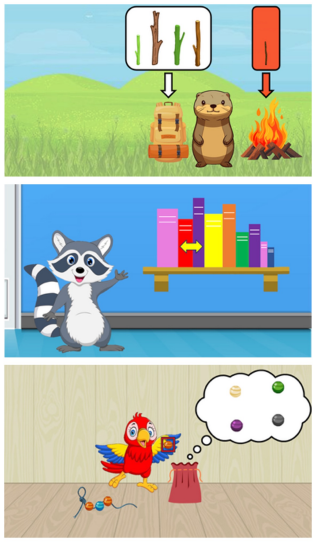

Figure 1: Illustrations from the the Animal Stories game. Top: Maximize (Otter); Middle: Sort (Raccoon); Bottom: Sample (Parrot).

Figure 1). Each game corresponded to an algorithm with a definite set of possible ending states. Critically, each story did not depict the algorithm being carried out in its entirety, as only the first few steps of the procedure were described before participants were asked to answer questions about it. Each story employed a series of comprehension checks to confirm that children understood the story as it progressed.

Each story was followed by six test questions, which fell into three broad categories: Questions 1 through 3 asked about the End State ("What will happen when the game is over?"), Questions 4 and 5 related to Efficiency ("Which of these starting states would be completed fastest?"), and Question 6 introduced a Novel Dimension ("If this game were played with a different feature than the one described in the story, what would happen?"). Following Questions 2 and 5 for each story (querying End State and Efficiency, respectively), the child was asked to explain their choice in their own words ("Why do you think that is? Just give me your best answer.").

Maximize (Otter). The Otter story encoded an algorithm for computing the maximum element in a set, disguised as a game played with the sticks in his backpack. At each step of the game, Otter removes a handful of sticks from his backpack, throws the shortest stick into the fire, and returns the rest to his backpack. Two steps were shown, then children were told that he continues until there is only one stick remaining. This algorithm will always result in the final stick being the longest—but this information was not included in the story.

For all choices and correct responses for all questions, see Figure 2. For the Otter story, Questions 1 through 3 (End State) each showed a set of sticks and asked, "If Otter has all these sticks in his backpack, which one will be the winner?" (4, 7, and 10 AFC, respectively). Participants simply needed to select the longest stick. Questions 4 and 5 (Efficiency) showed two backpacks with different numbers of sticks and asked, "Let's say Otter had two backpacks to choose from. If Otter wants to play the game quickly, which backpack should he choose?" (7 vs. 4, and 10 vs. 7). Participants needed to select the backpack with fewer sticks, as it would take fewer steps to remove all but the final stick. In Question 6 (Novel Dimension), the participant was shown a set of sticks where the longest stick was green. We asked, "Otter throws all of the green sticks into the fire. Then, he plays the game with only the brown sticks in his backpack. If Otter does this, which stick do you think will be the winner?" Participants had to ignore the green sticks and select the longest of the remaining (brown) sticks; this question was included to rule out a simple "always select the longest stick" heuristic.

Sort (Raccoon). Raccoon encoded an algorithm for sorting a sequence through iterative swaps of adjacent pairs, disguised as a game played with books on a shelf. Raccoon starts at the leftmost side of the shelf, compares the height of the first two books, and swaps their locations if the right book is taller than the left book. She then compares the height of the next two books, swapping them if necessary to have the taller book on the left. If any books swap positions, Raccoon starts again at the beginning. Three swaps were shown, followed by 2 comparisons that did not require swaps, then children were told that the game continues until Raccoon can go from the beginning to the end of the bookshelf without needing to swap any books. This algorithm results in the books being ordered by height, with the tallest on the left and the shortest on the right—but again, this end result was not described nor shown to participants.

Questions 1 through 3 (End State) showed a scrambled bookshelf and asked, "If we start with these books, which one of these shows what the bookshelf will look like after Raccoon plays the game?" (all 4 AFC). Participants needed to select the only option that depicted books in descending height order. Questions 4 and 5 (Efficiency) each provided two options of scrambled bookshelves where one bookshelf was more scrambled than the other (and would thus require many more swaps to order them) and asked, "If Raccoon wanted to finish the game quickly, which of these shelves would be best for playing the game?" The correct response was the less-scrambled bookshelf. Finally, in Question 6 (Novel Dimension), participants were told that Raccoon decided to play her game with a new set of grayscale books and would be swapping based on their color (from black to white) instead of their height. Participants were shown a scrambled bookshelf and asked, "If we start with these books, which one of these shows what the bookshelf will look like after Raccoon plays the game?" (4 AFC). Participants had to ignore the option where the bookshelf was arranged in descending height order, and instead select the one where the books were ordered from black to white.

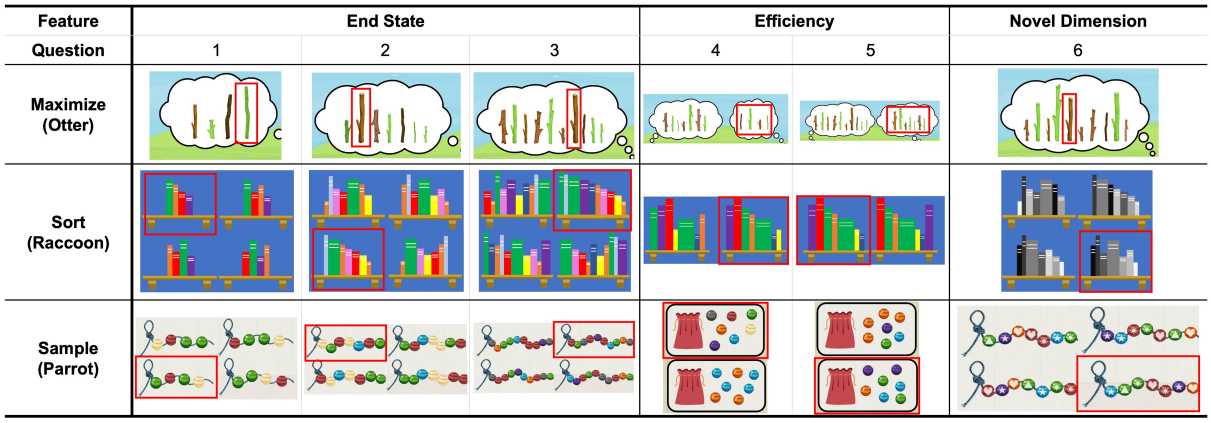

Figure 2: Response options for each forced choice question; red boxes indicate correct choices.

Sample (Parrot). The Parrot story encoded an algorithm for sampling without repetition, disguised as a story about putting beads onto a necklace. At each step, Parrot randomly pulls a bead out of her bag and compares it to the previous bead. If the beads are different colors, she places the new bead on the necklace. If the beads are the same color, she puts all of the beads back into the bag and starts again. Four samples were shown from a bag containing 8 beads, the last of which resulted in Parrot starting over, then 2 more samples were shown. Children were told that the game ends once there are no beads left in the bag. The end result of this algorithm is a chain of beads where no two consecutive elements are the same color.

In Questions 1 through 3 (End State), the participant was shown four possible bead configurations and asked, "Which necklace could Parrot make with this bag?" The correct response was the only necklace with no consecutive repeated colors. In Questions 4 and 5 (Efficiency), participants were shown two bags of beads, one of which has more beads of one color, and are asked, "Which of these bags would be best if Parrot wanted to play her game without having to start over too many times?" Participants had to select the option with fewer beads of one color. Finally, in Question 6, Parrot played her game with a new set of beads with shapes on them, using their shape rather than their color to determine placement. Participants were shown four necklaces and asked, "Which necklace could Parrot make with this bag?" Participants had to select the necklace where there were no adjacent matching shapes, rather than the one with no adjacent matching colors.

Procedure

Participants were tested either in lab or at a local children's science museum; all were tested individually. Informed consent from the child's parent or legal guardian was obtained prior to the start of the study. Children were asked if they would like to play our game, and were told that they could stop at any time if they did not want to continue. All test sessions were video recorded.

The experimenter read the three stories to the child in a fixed order (Otter, then Raccoon, then Parrot). Whenever the participant answered a comprehension check question incorrectly, the experimenter provided the correct answer (e.g., "Actually, this stick is the longest stick"), then repeated the question. They did not continue with the story until the participant responded correctly.

For the forced-choice test questions, the experimenter gave a pointer stick to the participant and told them to use it to point to their choice on the screen. Their choice was later coded from the video recording. For the free response questions, the child responded verbally and an experimenter later transcribed their response from the video recording.

The experiment took approximately 10 minutes to complete. For those who participated in lab, the guardian received $10 and the child received a toy. Participants who were tested at the museum were not paid and were instead given a sticker, in accordance with museum policy.

Results

Data

While the experiment included a total of 18 forced-choice test trials, some participants did not answer every question (two participants answered 16/18, while three answered 17/18). The final dataset included 1811 trials.

Children successfully reason about novel algorithms

First, we asked whether children responded correctly to each question from each story. We ran a series of binomial tests, separately for each question, comparing the number of children who responded correctly to the chance rate for that question with Bonferroni corrections across all tests. Children responded correctly at an above-chance rate to nearly every single question (corrected $ps < .001$; see Figure 3). As a group, children were not above chance for only two questions: the final two questions for the Raccoon story (corrected $ps = .327$; for further analysis on these two questions, see *Age effects*).

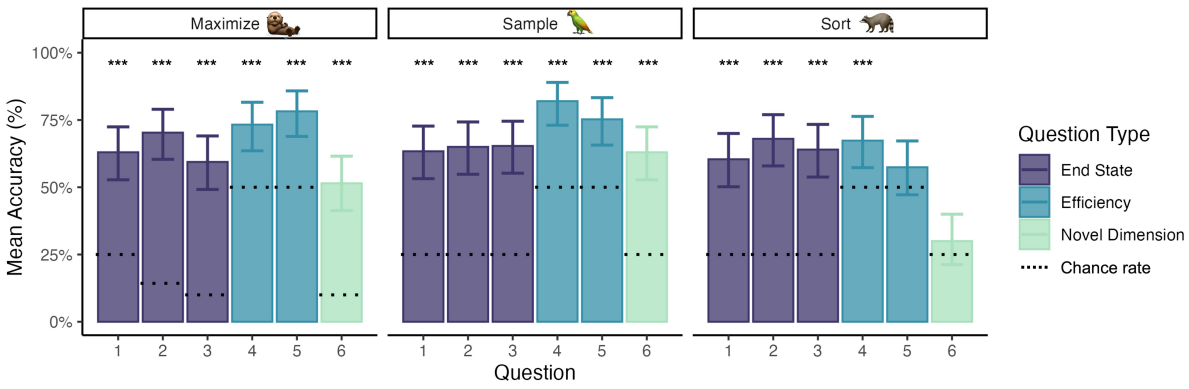

Figure 3: Average accuracy per question. As a group, children responded correctly to 16/18 questions. Error bars reflect 95% CIs from binomial tests; the dotted line represents the chance rate for each question, which varied depending on the number of options. $***p < .001$

Children correctly reasoned about multiple features of algorithms

Next, we asked whether children's ability to reason about novel algorithms was similar across different features: what End State would result if the algorithm were carried out, which of multiple starting states would be completed with the most Efficiency, and what would happen if the algorithm were applied to a Novel Dimension. We once again used binomial tests, separately for each feature type, with Bonferroni corrections across all three tests. We found that children responded correctly at an above-chance rate to questions about all features (End State: $\%_{correct} = 64.3\%$, $\%_{chance} = 22.1\%$; Efficiency: $\%_{correct} = 72.2\%$, $\%_{chance} = 50\%$; Novel Dimension: $\%_{correct} = 48.2\%$, $\%_{chance} = 20\%$; corrected $ps < .001$). This suggests that children's reasoning about novel algorithms is not shallow or based on simple heuristics, but instead utilizes a deeper, more flexible algorithmic representation that they were nonetheless able to derive from only a brief description.

Forced choice accuracy is predicted by explicit reasoning

Following six of the forced-choice questions, we asked children to explain why their previous choice. We coded their responses into six categories: 1) correct dimension and correct direction (e.g., "because it's the tallest"), 2) correct dimension and *incorrect* direction (e.g., "because it's the shortest"), 3) incorrect dimension (e.g., "because it has seven beads"), 4) irrelevant response (e.g., "because it's my favorite"), 5) "I don't know," and 6) no or unintelligible answer (which usually occurred due to the relatively noisy museum testing environment). We excluded responses with no or unintelligible answers from the analysis; this removed eight subjects, whose answers were all unintelligible, from the analysis ($N = 93$). We then collapsed this coding into a binary variable encoding whether or not the child provided the correct reasoning response (i.e., 1 if the child referenced the correct dimension and correct direction, 0 otherwise).

First, we asked whether subjects' likelihood of responding correctly to a given question was predicted by whether they provided the correct reasoning for that question. We performed a mixed effects logistic regression predicting accuracy from reasoning (binary), with participant ID was included as a random intercept. This analysis revealed that subjects were significantly more likely to respond correctly if they provided correct reasoning compared to any type of incorrect reasoning ($B = 2.17, z = 8.16, p < .001$; see Figure 4a). We then evaluated whether different types of incorrect reasoning yielded different accuracy rates, again using a mixed effects logistic regression predicting accuracy from reasoning, excluding correct reasoning. We found that subjects were less likely to respond correctly if they invoked the correct dimension but the wrong *direction* compared to all other kinds of incorrect reasoning (incorrect dimension: $B = 1.70, z = 2.63, p = .009$; irrelevant response: $B = 1.62, z = 2.53, p = .011$; "I don't know": $B = 1.32, z = 1.91, p = .056$). Interestingly, this suggests that an intermediate level of understanding—children who correctly identify the relevant dimension but misunderstand what direction to use—will actually lead to worse performance compared to a lower level of explicit understanding (e.g., "I don't know"), perhaps due to a higher reliance on random responding at lower levels of understanding.

Next, we investigated whether one's overall reasoning performance predicted one's overall performance on the forced-choice questions. To do this, we calculated each subject's average reasoning accuracy (across all questions for which they had an intelligible answer); we removed subjects who had fewer than 4 (of 6) intelligible answers from this analysis (removing $N = 15$ subjects). We used a simple linear regression to evaluate whether average accuracy on reasoning questions, age, and their interaction predicted forced-choice performance.

This analysis revealed that a higher reasoning score pre-

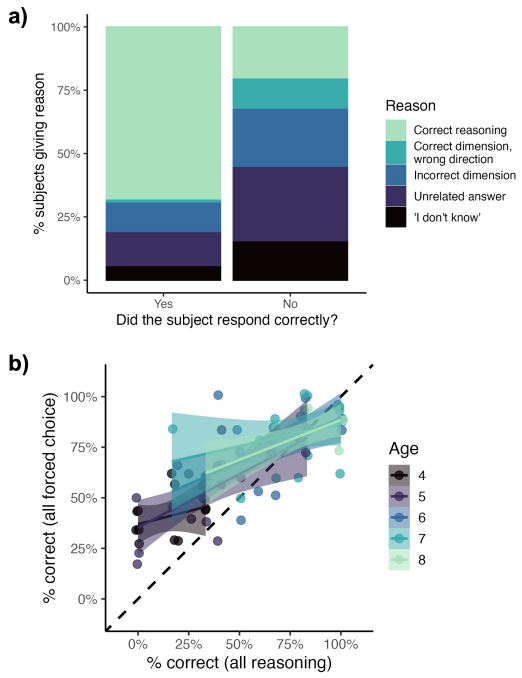

Figure 4: Explicit reasoning. a) Participants were more likely to provide correct reasoning following a correct forced-choice response. b) Explicit reasoning performance predicted forced-choice performance across all questions, even when accounting for age; each point represents one participant.

dicted higher overall accuracy on forced-choice questions ($B = .782, t = 2.856, p = .006$; see Figure 4b). Age also predicted accuracy even when accounting for reasoning ($B = .065, t = 2.182, p = .032$), and the interaction between reasoning and age was not significant ($B = -.057, t = -1.322, p = .190$). Taken together, these results indicate that children who were able to explicitly report the reasoning behind their choices were more likely to perform better—not only on the questions they were reasoning about, but on reasoning about algorithms in general.

Age effects

Performance improves with age. First, we investigated whether accuracy on the forced-choice questions varied with age, again using a mixed effects logistic regression. This revealed that performance significantly improved with age (years), $B = .578, z = 8.55, p < .001$ (see Figure 5a). We then evaluated whether each age group's performance was significantly above chance, again using a series of binomial tests with Bonferroni corrections. This revealed that children of all ages responded correctly at a rate higher than expected by chance, even the youngest (Overall chance rate: 31.1%; 4-year-olds: 41.3%, 95% CI = [36.1%, 46.6%]; 5-year-olds: 50.1%, 95% CI = [45.0%, 55.3%]; 6-year-olds: 72.8%, 95% CI = [68.3%, 76.9%]; 7-year-olds: 78.6%, 95% CI = [74.1%, 82.6%]; 8-year-olds: 81.3%, 95% CI = [76.0%, 86.0%]; all $ps < .001$).

As a group, participants were only at chance on two questions—Raccoon 5 and 6. We asked if the likelihood of responding correctly to these questions improved with age, again by comparing each age group's average performance on each question to chance. We found that 7-year-olds were significantly above chance on Raccoon 6 (corrected $p = .009$), but no other comparison reached significance (corrected $ps > .130$), suggesting that these particular questions were generally difficult, even for our oldest subjects.

The developmental trend is consistent across algorithms. Next, we asked whether the developmental trend was consistent across stories, as each story depicted a different algorithm. Because the different stories had different overall chance rates, we could not directly compare performance across them. Instead, we computed the average accuracy for each age for each story, then z-scored the average accuracies within each story. The resulting value then indicates how well each age group performed relative to the other age groups on each story. We ran a linear model predicting this standardized mean accuracy value from story, age, and their interaction.

Once again, there was a significant effect of age, $B = .583, t = 4.62, p = 0.001$. The effect of story on accuracy is not interpretable because the scores were standardized separately. Critically, the interaction term, which quantifies whether the age trend varied between stories, was not significant (single term deletion: $p = .893$; see Figure 5b). This indicates that children's relative ability to reason about each type of algorithm developed at a similar rate.

The developmental trend is consistent across algorithmic features. We asked whether the developmental trend differed across question types (End State, Efficiency, and Novel Dimension). Due to varying chance levels for each question type, we once again standardized the mean score for each age within each question type, then ran a linear model predicting the standardized mean accuracy from question type, age, and their interaction.

Here, age remained a significant predictor of accuracy, $B = .449, t = 2.695, p = .025$. Critically, the interaction between age and question type was not significant (single term deletion: $p = .597$; see Figure 5c). This suggests that children's ability to reason about these different features of algorithms also develops at a relatively similar rate.

Discussion

We found that children can make inferences about novel algorithms by as young as four years of age. The ability to reason about algorithmic processes starts to develop very early in life, and does not depend on first establishing procedural mastery. We suggest that the ability to reason about algorithms may be a domain-general ability that develops in tandem with the development of competency in carrying out such algorithms. Consistent with this interpretation, prior work demonstrates

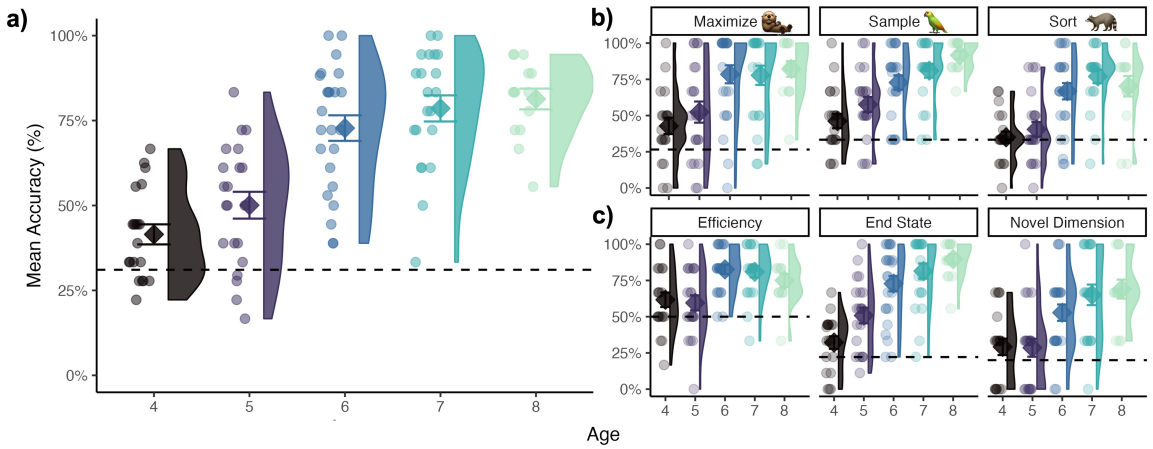

Figure 5: Average accuracy by age. a) Even the youngest children were above chance overall. b) The developmental trend did not differ between algorithms. c) The developmental trend did not differ between algorithmic features. Error bars reflect SEs on subject means, and dots are jittered on the x-axis.

that conceptual and procedural mastery often develop in tandem, iteratively providing support for one another's development (Rittle-Johnson et al., 2001).

While algorithms are ubiquitous in human life (e.g., mathematics, cooking, house repairs, etc.), past research has focused on how algorithmic skills develop in isolated domains. We propose that algorithmic reasoning be studied as a capacity of interest unto itself. Whether the ability to reason about algorithms derives from a distinct capacity, or instead relies on shared mechanisms with other aspects of cognition such as pattern detection (Wijns et al., 2019), relational reasoning, or rule-based inference remain open questions. The exact mechanisms by which children succeed at reasoning about algorithms will be fruitful avenues of future research. Candidate processes (consistent with the results presented here) include simulations, input-output mappings, and simpler heuristics (although note that our "Novel Dimension" questions were included to rule out extremely simple rules or heuristics, such as "Always pick the longest stick").

Regardless of the cognitive basis of these abilities, future work should investigate to what extent algorithmic reasoning capacity predicts one's ability to acquire procedural mastery with new algorithms. Children's intuitions about the function and efficiency of an algorithm may be leveraged to improve the speed and accuracy with which they carry out said algorithm. By studying domain-general algorithmic abilities, we may discover that cognitive processes are shared across domains that have analogous procedural structures.

Given that much of the content of basic mathematics–for example, counting, arithmetic, and geometry–involves acquiring and appropriately applying algorithms (Amalric et al., 2017; Piantadosi et al., 2012; Rittle-Johnson et al., 2001; Rule et al., 2020; Wijns et al., 2019), this algorithmic reasoning ability is likely to be essential to successful math learning. We propose that the study of algorithmic reasoning is likely to yield novel insights regarding the development of mathematical understanding. Future work in this area includes better understanding to what extent one's capacity to reason about algorithms predicts the accuracy or speed with which one masters said algorithms, and to what extent improving intuitive reasoning about algorithms improves algorithmic performance. By studying the domain-general ability to make inferences about novel algorithms, we will better understand what cognitive processes are shared across domains that have algorithmic content.

References

Amalric, M., Wang, L., Pica, P., Figueira, S., Sigman, M., & Dehaene, S. (2017). The language of geometry: Fast comprehension of geometrical primitives and rules in human adults and preschoolers (R. Gallistel, Ed.). *PLOS Computational Biology*, *13*(1), e1005273. https://doi.org/10.1371/journal.pcbi.1005273

Byrnes, J. P. (1992). The conceptual basis of procedural learning. *Cognitive Development*, *7*(2), 235–257. https://doi.org/10.1016/0885-2014(92)90013-H

Khemlani, S. S., Barbey, A. K., & Johnson-Laird, P. N. (2014). Causal reasoning with mental models. *Frontiers in Human Neuroscience*, *8*. https://doi.org/10.3389/fnhum.2014.00849

Kingma, J. (1987). Training of Seriation in Young Kindergarteners. *The Journal of Genetic Psychology*, *148*(2), 167–181. https://doi.org/10.1080/00221325.1987.9914546

Piantadosi, S. T., Tenenbaum, J. B., & Goodman, N. D. (2012). Bootstrapping in a language of thought: A formal model of numerical concept learning. *Cognition*, *123*(2), 199–217. https://doi.org/10.1016/j.cognition.2011.11.005

Rittle-Johnson, B., Siegler, R. S., & Alibali, M. W. (2001). Developing conceptual understanding and procedural skill in mathematics: An iterative process. *Journal of Educational Psychology*, *93*(2), 346–362. https://doi.org/10.1037/0022-0663.93.2.346

Rule, J. S., Tenenbaum, J. B., & Piantadosi, S. T. (2020). The Child as Hacker. *Trends in Cognitive Sciences*, *24*(11), 900–915. https://doi.org/10.1016/j.tics.2020.07.005

Wijns, N., Torbeyns, J., De Smedt, B., & Verschaffel, L. (2019). Young Children's Patterning Competencies and Mathematical Development: A Review. In K. M. Robinson, H. P. Osana, & D. Kotsopoulos (Eds.), *Mathematical Learning and Cognition in Early Childhood* (pp. 139–161). Springer International Publishing. https://doi.org/10.1007/978-3-030-12895-1_9

Wynn, K. (1989). Children's Understanding of Counting [ISBN: 9783540773405]. *Biennial Meeting of the Society for Research in Child Development*.

Yang, H. A., Thompson, B. D., & Kidd, C. (2025). Children spontaneously discover efficient solutions to a difficult sorting task. *Nature Human Behaviour*, *10*(1), 80–91. https://doi.org/10.1038/s41562-025-02302-6