Bootstrapping in Geometric Puzzle Solving

Xiangying He (yuhxy0717@outlook.com)
Department of Psychology, University of Edinburgh

Bonan Zhao
School of Informatics, University of Edinburgh

Neil R. Bramley
Department of Psychology, University of Edinburgh

Abstract

We explore how people "bootstrap", or reuse chunked action sequences, to tackle complex problems, in a novel puzzle task. In this task, participants perform sequences of actions to recreate target shapes. In our experimental condition, participants are trained on problems whose best solutions share a distinct abstract action sequence, or schema. Meanwhile, a control group trained on tasks of commensurate difficulty whose solutions did not conform to this pattern. We find experimental-condition participants outperform controls in a set of more difficult test puzzles whose solutions are compositional generalizations of experimental group's training tasks. Notably, the experimental group outperformed controls even in "far transfer" tasks that lacked surface similarity to training in both their target shape and solution sequence. Our results provide a compelling demonstration of the human ability to cache and reuse abstract patterns, offering new insights into how humans approach complex problems that, naively, seem to demand a prohibitive amounts of planning or trial and error.

Keywords: bootstrapping; problem-solving; chunking' schemata; planning

Introduction

A characteristic feature of human cognition is our ability to bootstrap: to flexibly reuse pre-existing concepts, skills or solutions as components in the creation of new complex concepts, skills or solutions. Understanding the cognitive and computational basis of bootstrapping seems key to the puzzle of how it is that we are able to invent and acquire complex concepts and skills despite our limited working memory and computational bandwidth.

Boostrapping in Learning

Bootstrapping has been studied in various cognitive domains, particularly in conceptual development and language acquisition. Carey (2004) explored bootstrapping in concept formation by examining how children combine their knowledge of counting sequences with their understanding of set sizes to 'construct' an abstract functional concept of numbers. Piantadosi et al. (2012) laid out a computational model of this phenomenon based on the Probabilistic Language of Thought framework, synthesizing this and other such insights as the result of recursively building upon primitive mental operations. Bootstrapping off of Carey and Piantadosi's work, Zhao et al. (2024) further identified, and demonstrated experimentally, that bootstrapping enables agents with limited computational resources to search deeply-nested concepts via principled cache-and-use, learning complex compound concept only if the examples are ordered in a way that allows the agent to first infer and cache useful sub-concepts.

In language acquisition, bootstrapping refers to the process by which children use one type of linguistic knowledge to help infer another. For instance, Pinker (1984) and Gleitman (1990) showed that children use semantic and syntactic contextual constraints to infer both grammar and word meanings. More recently, Yang and Piantadosi (2022) argued, contra Chomsky (1965), that bootstrapping enables language to be learned from the ground up.

Boostrapping in Problem-Solving

Like in learning concepts and languages, people also reuse previous solutions to solve later problems. Prior work has highlighted how individuals define and transfer action subsequences, or "moves", to support efficient strategy development. Catrambone (1998) emphasized the benefits of subgoal labeling in mathematical problem solving, showing that explicit decomposition supports schema formation and transfer. Interestingly, classical studies such as Luchins' (1942) water jug problem, also revealed how previously learned solution sequences can hinder flexibility when applied overrigidly to new tasks, a phenomenon known as the Einstellung effect (see also, Binz & Schulz, 2023).

We view bootstrapping in problem-solving as the process where learners extend their earlier ideas and (sub)solutions in the construction of more advanced problem-solving strategies (Tian et al., 2020; Zhou et al., 2024). By doing so, we align problem-solving with broader theoretical frameworks in cognitive science, such as the 'Language of Thought' (LoT) hypothesis, and classical theories of rule-based reasoning. The LoT hypothesis, popularized by Fodor (1975), posits that cognitive processes are structured like language, involving mental symbols with defined syntax and semantics. Similarly, symbolic reasoning theories described by Newell and Simon (2007) emphasize the manipulation of symbolic representations to enable logical thinking beyond concrete experiences.

This view is also connected with several established cognitive theories and research topics, including chunking (Gobet et al., 2001), creative hypothesis generation (Bramley & Xu, 2023), and analogy (Gentner, 1983). Chunking consolidates problem-solving steps into "chunks" of knowledge that can be frequently redeployed, reducing cognitive load and accelerating performance in settings where the steps are reusable

(Blessing & Anderson, 1996; Newell, 1990) . The fact that smaller chunks can be combined hierarchically into larger, more complex ones, captures one of the core ideas of bootstrapping. For example, Ho and Liausvia's (2013) study on incremental rule chunking for problem-solving exemplifies this process, showing how chunked rules can be used to tackle more complex problems by narrowing the search space and offering efficient sub-steps. Similarly, creative generation of new ideas is characterized by a process whereby we stochastically combine and recombine primitive elements into new representations (Lake et al., 2017). Tian et al. (2020) highlight the role of compositionality in developing structured internal representations that allow for flexible reasoning and learning in a character-drawing task, while Rubino et al. (2023) find people perform this kind of compositional reasoning even under time pressure. Moreover, the analogical reasoning literature records situations where people seem to recognize re-usable structure and associated insights and deploy it in analyzing new domains (Lovett & Forbus, 2017). For example, Kim et al. (2020) demonstrated that analogical reasoning can be used to solve Ravens Progressive matrix puzzles with few examples.

Despite the richness of boostrapping in problem-solving, much of the existing research on problem-solving has focused on specific aspects (e.g., strategy formation, knowledge transfer, or skill acquisition) and has rarely directly measured the level of bootstrapping with purpose-designed behavioral experiments. This lack of empirical evidence limits our understanding of the underlying mechanisms that enable individuals to solve complex problems. To address this gap, the present study investigates bootstrapping processes in an unfamiliar problem-solving context. Using a novel computer-based puzzle-solving task, we explore how individuals extract solution schema from simple tasks and apply them in more complex ones. We then argue that this paradigm holds promise for advancing research in human cognition and informing the development of adaptive AI systems capable of iterating toward solutions to complex problems.

The Geometric Puzzle Task

Task Interface and Rationale

We designed a computer-based puzzle-solving task to examine bootstrapping processes in problem-solving. Participants were required to construct target shapes using L-shaped pieces. They did this within a 9×9 grid where both x and y coordinates wrapped around, such that position $(\frac{N}{2}+1, \frac{N}{2}+1)$ was $(-\frac{N}{2}, -\frac{N}{2})$. Participants could combine the L-shaped pieces through sequential use of four operations: (1) Adding an upright puzzle piece in the center of the current view of the grid (pressing the "L" key), (2) Rotating everything 90 degrees clockwise ("R" key), (3) Flipping everything horizontally ("F" key), (4) Moving everything in a cardinal direction (arrow keys) (see Figure 1). Since moving the arrangement did not alter the shape or orientation, and solutions did not need to be centered within the grid to be correct, only the

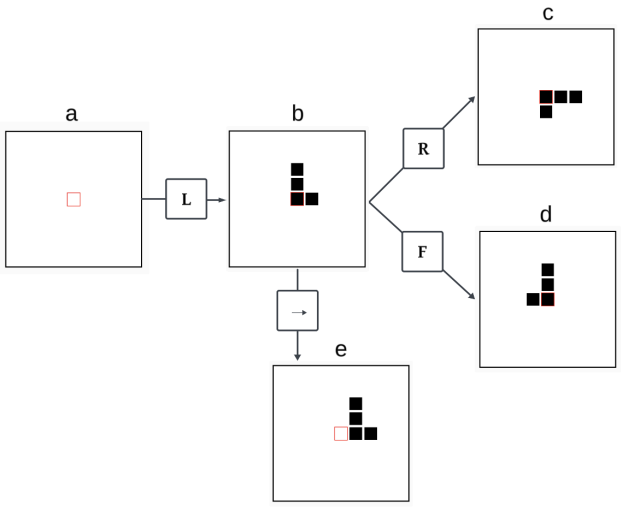

Figure 1: Task interface: (a) Empty grid, with red square indicating the center of the grid; (b) Pressing the "L" key to add an L-shaped piece to the center of the grid; (c) Pressing the "F" key to flip the L-shaped piece horizontally; (d) Pressing the "R" key to rotate the piece 90 degrees clockwise; (e) Moving the piece using arrow keys ($\leftarrow, \uparrow, \rightarrow, \downarrow$)

"L", "R", and "F" operations were counted as critical steps in later hypotheses and analyses. Crucially, all operations apply to the entire shape created so far, demanding careful forethought to produce a sequence that recreates a target pattern. As target shapes grow in complexity, the need for sequential strategy planning increases, elevating memory load and task difficulty. This task design provides a novel context with minimal reliance on prior knowledge, where participants must find ways to combine simple operations into complex solutions. This deceptively simple design enables precise measurement of learning, efficiency, as well as reuse.

Task Design: Training and Testing Phases

The experiment consisted of a training phase and a testing phase, manipulating the content of the Training phase in two between-subject conditions. Both conditions included five training tasks. During training, the experimental group completed three tasks where the target shapes involved two L-pieces (see Figure 2A). The optimal solution sequence required adding a L piece (key L), rotating the entire configuration (R), moving it (M), and repeating the addition (L) and rotation (R). Concretely, Task 1^e requires L+R+M$^{(\leftarrow\times 3, \uparrow\times 1)}$+L+R, Task 2^e L+R+L+R, and Task 3^e L+R+M$^{(\uparrow\times 2)}$+L+R. These tasks emphasized a critical solution schema: rotating the entire configuration after adding the second piece. This schema is key for solving more complex puzzles in the testing phase. We can represent this solution schema with a Regular Expression `(LRM*)+`, reading as one or more applications of a subsequence "L" followed by "R" and then any number of "M" steps. The control group

completed three tasks of similar complexity (see Figure 2B), where the optimal solutions involved flipping instead of rotating (Task 1^c: L+F+M$^{(\rightarrow\times2,\downarrow\times2)}$+L, Task 2^c: L+F+M$^{(\downarrow\times1)}$+L, and Task 3^c: L+F+M$^{(\rightarrow\times2)}$+L). This design ensured a similar level of mental manipulation difficulty without recursively manipulating the entire configuration. The number of black squares in the target shapes (and hence the amount of overlap of the pieces) was designed to match between groups. In addition, both groups completed two final training tasks (see Figure 2C) to ensure equal proficiency in rotation (R) and flipping (F) operations. Task 4 required rotating a single L-shaped piece three times, and Task 5 required flipping it once. Additionally, the interface included a "Clear all" button to allow participants to reset the grid if they made a mistake or got stuck. This button could be used as many times as needed; however, participants were required to solve each puzzle successfully before proceeding to the next one.

In the test phase, participants had a single opportunity to recreate each target shape without the "Clear all" button, requiring careful planning before starting. They could move to the next task by clicking a "Next" button, with an additional bonus for completing tasks using the minimal number of critical steps. Both groups completed the same four tasks. The tasks were designed to be of similar complexity (and to exceed the complexity of the training problems). Specifically, all test problems required the addition of least three L-pieces, resulting in a target shape composed of nine black squares (see Figure 3). Test task 1 could be solved most efficiently with the sequence L+R+M$^{(\uparrow\times2)}$+L+R+M$^{(\uparrow\times3)}$+L which reflected the "`(LRM*)+`" schema from the experimental groups's training tasks. Additionally, Task 1 included shape components familiar to both groups from their respective training Task 3. Task 2 employed the same optimal solution sequence (L+R+M$^{(\uparrow\times2,\leftarrow\times1)}$+L+M$^{(\leftarrow\times2)}$+L) but differed in the movement steps, meaning it did not involve reproducing or adding to a shape familiar from the training phase. Task 3 required a slight variation in the solution (L+R+R+M$^{(\uparrow\times2)}$+L+R+M$^{(\leftarrow\times3,\uparrow\times1)}$+L), requiring a third rotation, but crucially all still conform to other, slightly broader, schema such as "`(LR*M*)+`". Task 4 had a different optimal solution sequence: L+R+M$^{(\uparrow\times2,\leftarrow\times1)}$+L+F+M$^{(\uparrow\times1)}$+L, combining both rotation and flipping. This task maintained only an abstract structural similarity to the training tasks, following a recursive add-transform-move pattern where transformations could involve either rotation or flipping (i.e. a "`(L(R|F)*M*)+`" schema). The order of test tasks was randomized to control for practice effects.

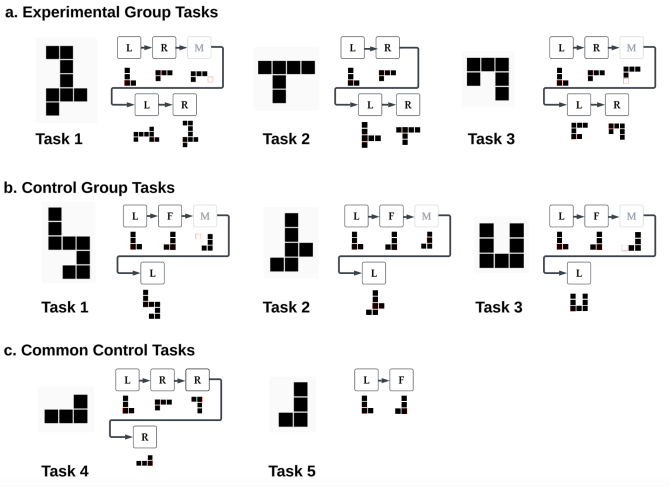

Figure 2: Training phase Tasks: (a) Experimental Group Tasks: 1^e-3^e (b) Control Group Tasks: 1^c-3^c (c) Common Training Tasks: 4-5.

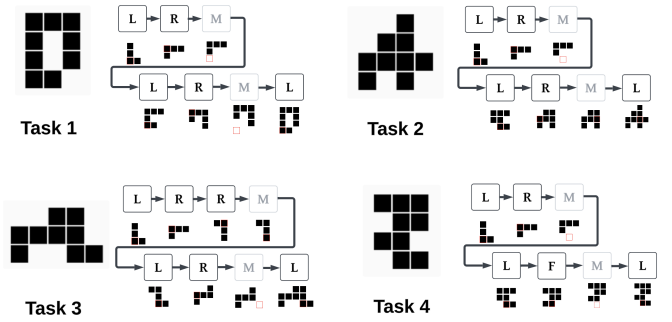

Figure 3: Test phase tasks

Hypotheses

Grounded in the theoretical framework of bootstrapping and its potential role in problem-solving, we propose the following hypotheses:

H1: The experimental group will demonstrate greater problem-solving efficiency than the control group in the test phase, as evidenced by (a) higher success rates, (b) fewer critical steps (L, R, F) to complete tasks, and (c) shorter completion time. This prediction assumes that the experimental group acquires a schema (e.g., the "`(LR+M*)+`", effectively a repeated addition-rotation strategy) during training (see Figure4). By reusing this schema, participants are expected to solve the more complex problem-solving structures in the test phase with greater efficiency.

H2: Within each group, we predict that participants will perform better on Test task 1 than on other test tasks, reflecting the principle of near-structure mapping in analogy (Gentner, 1983). Task 1 contains compound shapes familiar from both groups training tasks, potentially cuing them to build on whatever strategies they developed during training. However, we expected the experimental group to exhibit a smaller decline in performance across test Tasks 2-4, owning to their ability to generalize the schema acquired during training, potentially even to the farthest transfer required in Task 4.

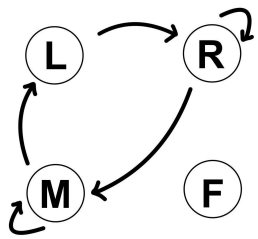

Figure 4: Schema formed by the experimental group in the training phase: a repeated rotation strategy (`(LR*M*)+`)

Methods

Participants

One-hundred and sixty-six participants (M age = 31.4, SD = 10.58; 77 females) were recruited from Prolific Academic. Four participants were excluded for low effort or missing data. The control group had 80 participants, and the experimental group had 82. Eligibility criteria included ages 18-55, UK or US residency, English as a first language, and a minimum education level of high school diploma or A-levels. Participants received a base payment of £1.80, with additional bonuses for correct task completion in the test phase (£0.30 per task and £0.50 for completing tasks using the minimum number of critical steps). The entire experiment, including instructions, tasks, and a brief questionnaire, took an average of 19.84 minutes (SD = 13.97).

Procedure

Participants were randomly assigned to either the experimental or control group. The experiment began with instructions on the task interface, followed by a hands-on practice session to familiarize participants with operations (L, R, and F keys). Participants were encouraged to add multiple L-pieces to explore these transformations. After passing a comprehension quiz, participants proceeded through the training and testing phases. Each phase was preceded by specific instructions and a corresponding comprehension check. After the tasks, the participants answered demographic questions and provided feedback on their experience, including engagement, perceived difficulty, and helpfulness of the training phase. A live demo is available at: eco.ppls.ed.ac.uk/ s2592856/exp/task.html.

Results

The results section is organized into five parts: (1) comparison of overall task completion between groups, (2) detailed analysis of problem-solving efficiency across tasks, (3) within-group performance comparisons using mixed models, (4) analysis of schema formation and reuse through sequence patterns and transition probabilities, and (5) subjective ratings of perceived helpfulness of the training.

Overall Task Completion

The experimental group significantly outperformed the control group in terms of overall task completion. A Mann-Whitney U test revealed that participants in the experimental group (Mdn = 1, IQR = 0–2.75) completed more tasks correctly than those in the control group (Mdn = 0, IQR = 0–1), $W = 2097$, $p < 0.001$, with a moderate effect size ($r = -0.36$) calculated by the rank-biserial correlation. Figure 5 displays the distribution of successfully completed tasks across groups.

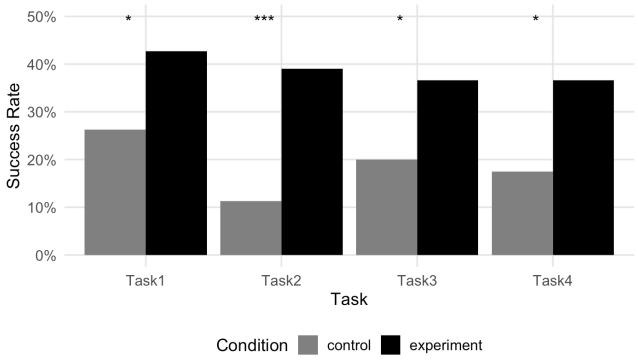

Figure 5: Comparison of success rate across tasks for Control and Experiment groups

Problem-Solving Efficiency for Each Task

To evaluate problem-solving efficiency, we analyzed performance metrics across tasks, including success rates, completion time, and the number of critical steps. Success rates were treated as the primary indicator, as they directly reflect performance accuracy. Completion time and critical steps were considered as supplementary measures, given the limited number of participants who completed the tasks correctly in both groups. All analyses were conducted after removing outliers using the interquartile range method (IQR).

χ^2 test results and effect size for task completion in control and experimental groups are presented in Table 1 . χ^2 tests revealed that the experimental group consistently achieved higher success rates across all four tasks compared to the control group (see Figure 5). The most pronounced difference was observed in Task 2, where the experimental group's higher success rate resulted in a large effect size ($h = 0.67$), while other tasks showed medium-to-large effects sizes ($h =$ 0.35–0.44).

There were no significant differences between the groups in completion time or the number of critical steps for most tasks, as indicated by t-tests and Mann-Whitney U tests. However, one exception occurred in Task 2. The t-test showed that the experimental group ($M = 21.45, SD = 10.01$) took significantly more critical steps than the control group ($M = 11.44, SD = 4.45$) to successfully complete the task, $t(27.04) = -3.70, p = 0.001, 95\% CI[-15.56, -4.45]$.

Table 1: Chi-Square Test Results and Effect Sizes for Task Completion in Control and Experimental Groups.

Task	Control (n, %)	Experimental (n, %)	χ^2	df	N	p-value	Effect size (h)
Task1	21(26.25%)	35(42.68%)	4.41	1	162	0.042	0.35
Task2	9(11.25%)	32(39.02%)	15.09	1	162	<0.001	0.67
Task3	16(20.00%)	30(36.59%)	4.69	1	162	0.030	0.37
Task4	14(17.50%)	30(36.59%)	6.52	1	162	0.011	0.44

Comparison of Performance Across Tasks in the Testing Phase

We used generalized linear mixed models (GLMMs) to assess whether performance on Task 1, which incorporated familiar shapes from training, was superior to performance on Tasks 2–4 within each group. The models were fitted using the `lme4` package in R (version 4.2.2), specifying a binomial distribution with a logit link function. The dependent variable was participants' binary task outcomes ∈(1:Success, 0: Failure). Task (Task 1 vs. Other) was entered as a fixed effect, and participant ID as a random effect. In the control group, performance on Task 1 was significantly better than that on other tasks ($\beta = -1.80, SE = 0.61, z = -2.96, p = 0.003$), with odds of success 6.1 times higher. No significant difference was found in the experimental group ($\beta = -0.33, SE = 0.31, z = -1.06, p = 0.29$). Random effects accounted for individual variation (control: SD = 7.53; experimental: SD = 1.61), and model fit indices supported the model adequacy (AIC/BIC: control = 227.30/238.60; experimental = 405.70/417.10).

Schema Formation and Reuse Patterns

We conducted a series of analyses to examine how participants formed and reused patterns acquired during training. First, regular expressions were used to detect transfer motifs in participants' action sequences. For near transfer tasks (Tasks 1–3), we applied the regular expression `(LRM*)+` to identify patterns involving one or more repetitions of the sequence "L" followed by"R" then zero or more "M"s. For the far transfer task (Task 4), we used the pattern `(L(R|F)M*)+`, allowing for "L" followed by either "R" or "F" and zero or more "M"s to capture more flexible reuse strategies. Across all tasks, the experimental group showed significantly higher schema reuse than the control group. In Task 1, a *t*-test revealed a significant difference between the experimental and control groups, $t(157.27) = -2.23$, $p = 0.027$. Similar significant differences were found in Task 2, $t(158.34) = -3.51$, $p < 0.001$; Task 3, $t(144.7) = -3.70$, $p < 0.001$; and Task 4, $t(149.64) = -3.69$, $p < 0.001$. These results suggest that the experimental group was more likely to apply schemas learned during training than the control group.

We further compared the predictive accuracy of participants' test-phase sequences using base action probabilities and transition probabilities derived from the training phase. In the experimental group, prediction based on training-phase transitions was significantly more accurate than predictions based on the overall action frequencies, $t(158.02) = -2.30$, $p = 0.023$. The same pattern emerged for the control condition, $t(153.76) = -3.33$, $p < 0.001$. Figure 6 displays the proportion of action transitions by condition and phase, highlighting that the L–R–M sequence was more prevalent in the experimental training condition than in the control condition.

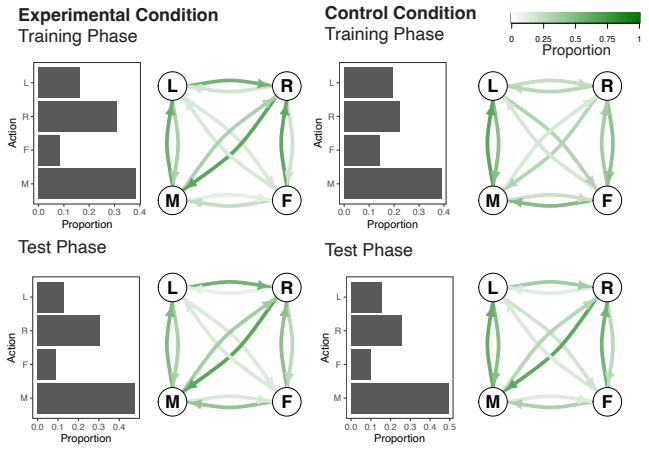

Figure 6: Proportions of action selections and transitions (omitting self transitions) by condition and phase.

Helpfulness

At the end of the experiment participants rated the perceived helpfulness of the training phase for completing the testing phase tasks on a 1-5 scale, with higher scores indicating greater perceived utility. Results showed that participants in the experimental group rated the training as significantly more helpful than those in the control group (Mdn = 3.62 vs. 3.01, $U = 2217$, $p = 0.002$). A mediation analysis using the *lavaan* package in R showed that success rate partially mediated the effect of training condition on perceived helpfulness (*indirect effect*: $b = 0.31$, $SE = 0.09$, $z = 3.40$, $p = .001$, 95% CI [0.15, 0.51]; *direct effect*: $b = 0.30$, $SE = 0.15$, $z = 2.00$, $p = .046$, 95% CI [–0.01, 0.57]). The total effect was also significant ($b = 0.61$, $SE = 0.16$, $z = 3.88$, $p < .001$, 95% CI [0.28, 0.92]). All coefficients reported were standardised. These findings indicate that although success rate partially

mediated the effect of training condition on perceived helpfulness, the significant direct effect suggests that the training itself contributed to perceived helpfulness beyond performance outcomes.

Discussion

Our analysis of schema formation and reuse patterns revealed that participants in the experimental group reused the schema formed during training significantly more frequently than those in the control group across all test phase tasks. This indicates that they were able to effectively construct and generalize a problem-solving schema from a limited number of training trials and apply it to solve both familiar and novel complex problems. These findings are consistent with previous work. For example, Tian et al. (2020) demonstrated that individuals can rapidly acquire abstract procedures that support generalization, while Lake and Piantadosi (2020) emphasized humans' capacity to learn and reason about algorithmic abstractions from limited data. Importantly, the predictive accuracy analysis showed that transition probabilities derived from the training phase were better predictors of participants' test-phase actions than simple action frequencies. This suggests that the experimental group's superior performance was not merely due to familiarity with rotation operations (i.e., R key usage) but rather reflected the successful formation and flexible reuse of the problem solutions at an abstract "schema" level.

Additionally, the experimental group achieved superior performance on tasks that involved the reuse of solution structures even at an abstract "schema" level. This improvement was not restricted to specific operation sequences but reflected a broader capacity to apply learned problem-solving strategies (the formed schema) to both familiar and novel tasks. This demonstrates the power of bootstrapping in enabling the development of flexible, adaptable cognitive strategies that can be applied to a wide range of increasingly complex problems. Notably, the consistent reuse of the `(LRM*)+` pattern in Tasks 1-3 and the adaptation of the schema `(L(R|F)M*)+` in Task 4 suggest that participants in the experimental groupwere able not only to transfer the schema but also to flexibly modify it to handle diverse tasks.

Contrary to our hypothesis, no significant differences were observed between the experimental and control groups in terms of completion time or the number of critical steps, except for Task 2. In Task 2, the control group completed the task using fewer steps on average. This finding is likely attributable to the high overall task difficulty, as reflected by the low success rates (40% for the experimental group, 20% for the control group) and participants' subjective ratings of task difficulty (average score around 8.5 on a 0-10 scale). Despite being instructed to minimize steps, participants appeared to prioritize task completion over step optimization. The control group's apparent efficiency in Task 2 is likely explained by the small number of successful participants ($n = 9$), who potentially possessed above-average spatial problem-solving abilities, thereby complicating direct comparisons between groups.

Comparing performance across tasks within each group, we found that the experimental group showed no significant difference in success rates across all tasks, whereas the control group performed significantly better on Task 1 compared to the others. Task 1 featured a familiar target shape from training. For the control group, this familiarity appeared to facilitate problem-solving. In contrast, the experimental group's consistent performance across tasks was likely driven by the reuse of the schema (i.e., an overall rotation strategy) formed during training, rather than shape recognition or memorization. These findings challenge traditional views of chunking, which typically emphasize its role as a memory aid for specific problems (Sakai et al., 2003; Servan-Schreiber & Anderson, 1990; Thalmann et al., 2019; Wu et al., 2023). Our results suggest that in addition to forming reusable chunks, bootstrapping also involves the abstraction of schemas. Unlike transferring learned chunks—which relies on the recognition of identical elements, bootstrapping fosters the creation of broader, more flexible cognitive structures, as testified in the cognitive development literature (Carey, 2004; Rule et al., 2020). Furthermore, the experimental group's lack of a performance boost on Task 1 (compared to other tasks for the experimental group), may imply a trade-off between generalizable problem-solving and task-specific familiarity.

Our study also has several limitations that invite future investigations. The moderate success rate we observed is likely due to the high difficulty of the testing phase, especially in the control group, which may have limited our statistical power. Increasing the sample size or adjusting the test phase task difficulty, such as by allowing multiple attempts, could help mitigate this. Future studies could also explore the metacognitive aspects of problem-solving, such as how quickly individuals recognize unsolvable tasks, or how time constraints influence the bootstrapping process, roviding further insights into how strategies are formed and adapted in dynamic contexts. Moreover, we hope to investigate group problem solving settings where recognition of successful strategies in others' may help parallelize the search problem for good solutions, while collaborative problem solving might foster division of labor and specialization (Almaatouq et al., 2021).

In sum, our study adds to the literature on problem solving (Gobet et al., 2001; Laird et al., 1984), contributing a novel paradigm that allows the measurement of compositional reuse. This task reveals how learners bootstrap strategies from simpler components, shedding new light on theories of bootstrapping in cognitive development.

References

Almaatouq, A., Alsobay, M., Yin, M., & Watts, D. J. (2021). Task complexity moderates group synergy. *Proceedings of the National Academy of Sciences*, *118*(36), e2101062118.

Binz, M., & Schulz, E. (2023). Reconstructing the einstellung effect. *Computational Brain & Behavior*, *6*(3), 526–542.

Blessing, S. B., & Anderson, J. R. (1996). How people learn to skip steps. *Journal of Experimental Psychology. Learning, Memory, and Cognition*, *22*(3), 576–598.

Bramley, N. R., & Xu, F. (2023). Active inductive inference in children and adults: A constructivist perspective. *Cognition*, *238*, 105471.

Carey, S. (2004). Bootstrapping & the origin of concepts. *Daedalus*, *133*(1), 59–68.

Catrambone, R. (1998). The subgoal learning model: Creating better examples so that students can solve novel problems. *Journal of Experimental Psychology: General*, *127*(4), 355–376.

Chomsky, N. (1965). *Aspects of the theory of syntax*. MIT press.

Fodor, J. A. (1975). *The language of thought*. Crowell.

Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. *Cognitive science*, *7*(2), 155–170.

Gleitman, L. (1990). The structural sources of verb meanings. *Language Acquisition*, *1*(1), 3–55.

Gobet, F., Lane, P. C., Croker, S., Cheng, P. C., Jones, G., Oliver, I., & Pine, J. M. (2001). Chunking mechanisms in human learning. *Trends in cognitive sciences*, *5*(6), 236–243.

Ho, S.-B., & Liausvia, F. (2013). Incremental rule chunking for problem solving. *2013 BRICS Congress on Computational Intelligence and 11th Brazilian Congress on Computational Intelligence*, 323–328.

Kim, Y., Shin, J., Yang, E., & Hwang, S. J. (2020). Few-shot visual reasoning with meta-analogical contrastive learning. *Advances in Neural Information Processing Systems*, *33*, 16846–16856.

Laird, J. E., Rosenbloom, P. S., & Newell, A. (1984). Towards chunking as a general learning mechanism. *AAAI*, 188–192.

Lake, B. M., & Piantadosi, S. T. (2020). People infer recursive visual concepts from just a few examples. *Computational Brain & Behavior*, *3*(1), 54–65.

Lake, B. M., Ullman, T. D., Tenenbaum, J. B., & Gershman, S. J. (2017). Building machines that learn and think like people. *Behavioral and Brain Sciences*, *40*, e253.

Lovett, A., & Forbus, K. (2017). Modeling visual problem solving as analogical reasoning. *Psychological Review*, *124*(1), 60–90.

Luchins, A. S. (1942). Mechanization in problem solving: The effect of einstellung. *Psychological Monographs*, *54*(6), i–95.

Newell, A. (1990). *Unified theories of cognition*. Harvard University Press.

Newell, A., & Simon, H. A. (2007). Computer science as empirical inquiry: Symbols and search. In *Acm turing award lectures* (p. 1975). Association for Computing Machinery.

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.

Pinker, S. (1984). *Language learnability and language development* (2nd). Harvard University Press.

Rubino, V., Hamidi, M., Dayan, P., & Wu, C. M. (2023). Compositionality under time pressure [preprint]. *PsyArXiv*.

Rule, J. S., Tenenbaum, J. B., & Piantadosi, S. T. (2020). The child as hacker. *Trends in cognitive sciences*, *24*(11), 900–915.

Sakai, K., Kitaguchi, K., & Hikosaka, O. (2003). Chunking during human visuomotor sequence learning. *Experimental brain research*, *152*, 229–242.

Servan-Schreiber, E., & Anderson, J. R. (1990). Learning artificial grammars with competitive chunking. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *16*(4), 592.

Thalmann, M., Souza, A. S., & Oberauer, K. (2019). How does chunking help working memory? *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *45*(1), 37.

Tian, L. Y., Ellis, K., Kryven, M., & Tenenbaum, J. B. (2020). Learning abstract structure for drawing by efficient motor program induction. *arXiv*.

Wu, S., Éltető, N., Dasgupta, I., & Schulz, E. (2023). Chunking as a rational solution to the speed–accuracy trade-off in a serial reaction time task. *Scientific reports*, *13*(1), 7680.

Yang, Y., & Piantadosi, S. T. (2022). One model for the learning of language. *Proceedings of the National Academy of Sciences*, *119*(5), e2021865119.

Zhao, B., Lucas, C. G., & Bramley, N. R. (2024). A model of conceptual bootstrapping in human cognition. *Nature Human Behaviour*, *8*(1), 125–136.

Zhou, Y., Feinman, R., & Lake, B. M. (2024). Compositional diversity in visual concept learning. *Cognition*, *244*, 105711.