ABSTRACT

PUGH, ZACHARY HARRIS. Left to Their Own Devices: Examining User Strategies in the Selection and Creation of Digital Tools. (Under the direction of Drs. Douglas J. Gillan and Chang S. Nam).

Technology is central to human experience. However, theoretical frameworks and studies of user behavior often treat technologies as pre-established conditions, whereas users in the real world make decisions beyond the scope of usage to include a tool's recruitment over alternative options, its design and customization, and its coordination with encompassing systems of artifacts and methods curated by the user. Assuming this broader approach of users as system builders, I present two studies examining people's strategies in the selection of smaller-scale usable resources to build larger-scale usable work systems. In Experiment 1, participants completed a task mimicking visual programming in which they built formula-calculating machines using resources that varied in a tradeoff between flexibility (ease of reconfiguring the resource's function) and stability (a resource's ability to persist without maintenance). Correspondingly, the elements of each problem either varied or remained the same across trials, entailing an optimal strategy in resource allocation. However, few discovered and implemented this strategy. In Experiment 2, participants could complete part of the computation using mental arithmetic as an alternative to a solely external computation. It was found that regardless of problem difficulty, participants generally preferring this "cognitive onloading," and most participants perseverated in their strategy regardless of changes in difficulty. These and other findings highlight a need to further explore how people craft strategies and systems through their artifacts, not just their perceptual and motor routines.

Keywords: Decision-making, human-computer interaction, customization

Left to Their Own Devices: Examining User Strategies in the Selection and Creation of Digital Tools

by
Zachary Harris Pugh

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Psychology

Raleigh, North Carolina

2025

APPROVED BY:

Dr. Douglas J. Gillan Co-Chair of Advisory Committee	Dr. Chang S. Nam Co-Chair of Advisory Committee
Dr. Jing Feng	Dr. Anne C. McLaughlin

BIOGRAPHY

Zachary Pugh graduated from the University of North Carolina at Chapel Hill in 2011 with a Bachelor of Arts in Anthropology. After serving four years in the Army as a logistics officer, he became engrossed in psychology through volunteer research at the University of North Carolina at Greensboro, and in 2017 he began his doctoral work in psychology with a concentration in Human Factors and Applied Cognition at North Carolina State University. Zach's research focuses on cognitive tools and the ways people organize artifacts to aid their own work, reasoning, and decision-making. Additionally, in various collaborations he contributed to the neuropsychology of emotion, neuroergonomics, and the design of experimental virtual environments.

ACKNOWLEGMENTS

To my family, thank you for your love and support. A special thanks to my “play testers” Teddy and Josey.

To my friends in the Human Factors Program, thank you for the fun times, fascinating conversations, and challenging projects. Poe 700 will always have a special place in my memory.

To Doug, thank you for your wisdom and patience and for accompanying me on many philosophical tangents.

To CS, thank you for introducing me to your lab and helping me navigate the daunting yet exciting waters of neuropsychology.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Introduction

Tool use makes up a cornerstone of the human experience, and psychology has studied this from several angles, ranging from basic research examining tool competency in humans and other animals (Mangalam et al., 2022) to applied research seeking how best to adapt contemporary tools to the physiological and psychological constraints of the human user (Norman, 2013). However, there is more to tool use and technology than strictly *use*. Tool use involves problem solving at multiple scales. Users construct and adopt tools, learn strategies of use and manipulation, customize tools, and coordinate them with other tools, artifacts, and the environment. Bødker and Klokmose (2011) use the term "artifact ecology" to refer to this broader scale. So people are not simply tool users; they are builders of work systems that integrate themselves and these artifacts.

This systems scope entails a few questions: How do users decide on what tools to integrate into their work system? Are users rational, or are there biases in their efforts to re-design and customize? Does the existing environment influence the workflow a user develops, and does a learned workflow influence future decisions an agent makes in crafting their work system? Many of these questions already have answers, but their answers come from research models that examine particular facets of system building in piecemeal, and so it stands to wonder whether and how well they generalize to conditions in which all facets are simultaneously under the user's control. These include the processes of learning perceptual-motor and cognitive strategies, of technology adoption, of technology customization, and of coordinating the tool within a larger-scale functional system. The goal of this study is to examine user behavior in an environment where the user negotiates these problems simultaneously.

Users develop techniques

For any given system, whether it is a single widget in a vacuum or a complex assortment of devices, there is the question of how best to carry out the task it is meant for. For instance, when cooking, do you cut the vegetables first, or start by preheating the oven? What if the heat source is a stovetop? The common culinary adage of *mis en place* suggests having materials already in place prior to initiating any time-sensitive processes. Similarly, do you memorize the steps in the recipe before beginning or alternate between reading instructions and following them? In this sense, good use strategy requires understanding what actions are possible or necessary, the sequence and timing of those actions, and the cognitive, sensory, and motor demands they entail.

Evidence suggests that people are indeed sensitive to the perceptual, motor, and cognitive demands of a task, enough to seek out strategies that optimize performance on a scale of milliseconds. Gray and Boehm-Davis (2000) examined data from a usability task in which participants clicked on blocks appearing on a screen, with blocks alternating between a fixed "home" position at the bottom of the screen and a random "target" position above. Because the home position was fixed, participants could adopt an anticipatory strategy in which they move the cursor immediately to the home button without visual search. The 150 ms latency difference between the target-to-home and home-to-target trials suggests that participants indeed adopted this strategy. In this and other tasks, Gray et al. (2006) have found that at a temporal scale of 0.3 to 3 seconds, people seek to minimize costs to time rather than to cognitive or motor resources.

However, in other cases the priority is to minimize effort (Walton et al., 2006). This sensitivity for optimization extends to cognitive actions as well, such as when solving arithmetic problems (Neth and Payne, 2001). Kool et al. (2010) found biases toward strategies that minimize cognitive demand, even when more efficient strategies are available.

Users adopt tools

One limitation on the study of perceptual-motor strategies is that the tasks or technologies participants engage in are typically a priori conditions of the experiment, at the expense of studying the processes in which users modify their conditions by changing the environment or acquire technology. To borrow from the literature on manual tool use, Osiurak et al. (2020) frame human tool use in terms of problem solving, which includes reasoning about the physical properties of the task to determine a desired action, inferring the kind of tool necessary to achieve the result, and then searching for the tool. In the context of information technology, the acquisition process is studied under models of technology acceptance (Lai, 2017). Perhaps the most prolific of these is the Technology Acceptance Model (TAM), which has a complicated theoretical lineage involving many variants of itself, as well as theoretical predecessors in psychology (Marangunic´ & Granic´, 2015). In short, the TAM states that the use of a system depends on two constructs—*perceived usefulness* and *perceived ease of use*—that mediate between the properties of the information system and the decision to accept that system. More specifically, these constructs directly influence an individual's intention to use, which in turn determines actual usage. Additionally, perceived ease of use has its own direct influence on perceived usefulness. The two constructs of TAM have been empirically validated (Davis, 1989) and applied extensively to empirical studies on technology acceptance.

Users adapt tools

A third degree of freedom is that technology itself is often amenable to modification and redesign by the user. Even when end users a far removed from the design cycles that created those tools in the first place, there is variety in just how much a user might be able to alter to the artifact's interface or even mechanism of operation. In some cases, this opportunity for modification is built into the artifact (e.g., reconfiguring key bindings in a game to improve performance). In others, it requires some external innovation at the user's end (e.g., developing an external script to modify controls). In yet others, customizability is designed out of the system to preserve the integrity of the device (e.g., code obfuscation).

With respect to human-computer interaction (HCI), this added degree of freedom falls under the generic term *customizability* (Mackay, 1990) and the more siloed term *end-user development* (Barricelli et al., 2019). In her thesis study of software customization, Mackay (1990) conducted interviews of Unix users and examined their customization records, with a few takeaways. Customization is a co-adaptive process, in that users and their software are mutually adapting. Customization tends to occur early in use and diminishes with time. It is often used for both exploration and encoding routine activities. User's customization reflect a sensitivity to both work context and external events. Users display a degree of inertia when software changes, potentially forfeiting the new software to maintain their old environment and its customizations. Finally, Mackay also observed a communal aspect to customization, with newcomers inheriting customization files prepared by predecessors.

Customization also has performance implications; in one experiment, Burkolter et al. (2014) had participants conduct a control process task using either a default interface or an interface that users were free to customize. Despite no differences in cognitive load and situation awareness, the group who customized their interface committed fewer errors and reported a greater degree of acceptance. Aside from pragmatic goals, users also customize to express a sense of identity and control (Marathe & Sundar, 2011).

While end-user development might be equated to customization, in practice its research tends to focus less on tailoring software and more towards implementing bespoke design systems such as scripting languages that the end user might use to write modifications (Barricelli et al., 2019). For instance, users of Unity game development software might use C# to develop games, but they may also use it to augment the software with bespoke editing tools to support the game design process.

Users Integrate Tools into Systems

Technique learning, acquisition, and customization could well be observed by placing a single user and tool in a vacuum, as controlled studies often do. But no tool is an island. A device's role and value depend on its coordination with other artifacts and the environment, which are themselves amenable to use and modification. A couple of frameworks have pushed for this ecological framing of technology use.

One of these is activity theory, which interprets behavior as an essentially goal-oriented and socially situated phenomenon (Engestrom, 2000). Behaviors are thus organized hierarchically according to the scale of their function: *Activities* at the highest level characterized by *motivations*, conscious *actions* at the intermediate level characterized by *goals*, and *operations* and the lowest level characterized by automatic and unconscious processes (Bødker and Klokmose, 2011). With respect to HCI and human factors, most attention is on actions and their constituent operations. For instance, a typist using a computer is an action, while their routinized motor movements on the keyboard and the computer's internal mechanism are both operations (Engestrom, 2000). Activity theory also emphasizes a fixed set of characteristics of an action: the actor, the object, instruments mediating the action, and the social context (i.e., rules, community, division of labor). Bødker and Klokmose (2011) extend activity theory to examine *artifact ecologies*, essentially emphasizing that a user's interaction with an artifact is situated within a broader system of artifacts.

Another systems-level framework is distributed cognition, which views cognition as a process that is not strictly in the head but rather allocated among minds (internal representations) and objects in the environment that serve as representations and computational mechanisms (external representations). For instance, the representations and inferences needed for maritime position fixing might be distributed among map plotters, bearing recorders, alidade operators, and the maps, rulers, and other representational instruments that they use (Hutchins, 1995). Distributed cognition therefore takes as its unit of analysis any system of minds and artifacts that have some role in the manipulation and propagation of representations. This framework has yielded significant findings in how people recruit artifacts and the environment to alleviate cognitive processes or even supplant them, and it has also given some insightful language for describing ecological cognition.

For instance, in the game of Tetris, players control the placement of descending blocks on a screen and must recognize whether a block will fit into unique spaces at the bottom of the screen. Kirsh and Maglio (1994) observed that, as an alternative to mentally rotating the block to test its fit, advanced players would manually rotate it rapidly using the controller. The resulting physical rotation can then be visually compared to the cavities at the bottom of the display to determine the block's fit. Physical action can therefore have a cognitive function. In this regard the authors proposed a distinction between "pragmatic actions", which bring the world state closer to an agent's goal, from "epistemic actions" that simply give the agent more information. A similar concept is cognitive offloading—"the use of physical action to alter the information processing requirements of a task so as to reduce cognitive demand" (Risko & Gilbert, 2016, p.

676). These notions of altering the environment and cognitive offloading circle back to earlier framings of perceptual-motor strategy development and customization.

A third and the oldest systems-level perspective here is cybernetics, the study of "control and communication in the animal and the machine" (Wiener, 1948). It is concerned with any kind of self-regulating system, with special interest in control structures such as regulatory feedback loops that help stabilize critical variables within a system, such as body temperature. These systems range from simple thermostats to biological life, and recently cybernetics has been extended, under the title of practopoiesis, as a model for biological and artificial intelligence (Nikolić, 2015). For this study, cybernetics provides a heuristic in guiding the design of the experimental tool-making environment.

Using Cybernetics to Examine System Building

Because the purpose of this study is to examine how users adapt their systems to changing demands, cybernetics is an especially appropriate framework. Specifically, the design of this study is centered on the Ashby's (1958) principle of variety. The Law of Requisite Variety states that to successfully maintain a variable at a target state amid some disturbance, a regulator must possess at least as many unique states as the disturbance can present. That is, a good regulator has a measured response for every possible kind of disturbance. For instance, consider how your wardrobe matches the temperature variation of your geographic region. The low temperature variety of arctic and tropical zones may be reflected in a low variety of clothing, while a temperate zone that varies widely across the seasons demands a wider variety in clothing with respect to insulation.

Extending from the concept of variety, two high-level but relevant concepts for this study are flexibility and stability, as well as the tradeoff between them. Flexibility refers to a system's ability to adapt to a wide (vs. narrow) variety of conditions. Compared to inflexible resources, adapting a flexible resource to a new condition should be easier. Stability refers to a system's resilience to perturbation. Consequently, stable resources continue functioning in the absence of continued coordination from the user's end, compared to unstable ones. Like cybernetics itself, flexibility and stability do not belong to a particular domain but are used loosely to describe a range of phenomena, including a flexibility-stability tradeoff in human goal-oriented behavior (Dreisbach & Fröber, 2019) and the stability-plasticity dilemma of biological systems (Nikolić, 2015).

How are these properties realized in technology? Take information storage media as an example. A map is easily drawn in the sand but is liable to erode, whereas a map in stone is resilient to weathering but tremendously difficult to engrave. The physical media of early writing systems appear to balance both properties. Clay is malleable enough to afford manual indentations using a reed, but it is also stable enough to persist when left alone, making it an ideal substrate for Babylonian Cuneiform. Cordage is both portable (variety over geographic space) and physically flexible, properties that the Inca Empire took advantage of in the form of quipu, interconnected strands of cordage that stored information in the form of knots (Ascher, 2008). The ability to tie and untie knots provided an expedient means for incrementing and decrementing tallies (Cajori, 1993). Pencil enables both writing and erasure, while pen enables only writing, yielding a more stable system in the sense that markings cannot be erased, accidentally or otherwise. In his study of a navigation team aboard a Navy amphibious assault ship, Hutchins (1995) observed that whereas most routes were plotted in pencil, a pen would be used to plot routine routes that were invariant from case to case, as this alleviated work from the navigation team during moments of high cognitive load. These innovations reveal a sensitivity to stability and flexibility that materials provide, although it is not clear how these innovations developed.

Related to system-building, flexible and stable resources may also be strategically coordinated if the task is complex enough, especially when an agent's executive or motor resources are limited. For instance, when bearded capuchin monkeys use a stone to crack a palm nut, they often begin by placing the nut onto an anvil stone prior to striking it. The nut's placement is not arbitrary, but it is often seated within a depression of the anvil stone, such that it cannot roll away once left alone. This stabilization enables the monkey to use both arms in their manipulation of the hammer stone. Similarly, asymmetric bimanual tasks (e.g., flint knapping) typically involve the use of one hand to stabilize an object and the other hand to carry out some action on it (Guiard, 1987; Stone et al., 2013). This dynamic explains distributed cognition as well. Hutchins (2005) refers to external representations, such as maps, as "material anchors" in the sense that they stabilize elements of a representation so that attention can be directed to more dynamic aspects of the cognitive task, such as mentally simulating the movement of elements on the map.

This is not to say that flexibility and stability gives a complete account of technology. The point is that these technologies and methods reflect the intrinsic demands tasks they are recruited for, which suggests that their culmination depended on some sensitivity of the user, designer, or both towards the properties of materials available and the demands of the task at hand. For this reason, the tradeoff between these two properties is an appealing starting point to test people's strategies, preferences, and biases in the creation of functional work systems.

Current Study

The goal of the present study was to examine people's strategies and biases during the process of system building in a digital environment. In contrast to the common paradigm of observing behavior in the context of fixed technology, participants built and customized their own systems from available resources. To make such a broad topic tractable, the study specifically examined how people prioritized flexible versus stable resources, programmed in such a way that the tradeoff between the two entailed an optimal building strategy for comparing behavior. To investigate this, a task was built with several desiderata in mind:

1. The task environment contains lower-order tools which are used to create a higher-order tool.
2. The lower-order tools feature either flexibility or stability but are otherwise equivalent in their user interactions.
3. There is an efficiency tradeoff between flexibility and stability, such that flexible resources are favorable in volatile conditions and stable resources are favorable in static conditions.
4. The task has intrinsic demands for both flexibility and stability.
5. The task's demands for flexibility vs. stability can change over time, adding a second layer of dynamics to study the participant's adaptation.

To satisfy these, a task mimicking visual programming was developed in which participants built arithmetic calculators using two types of resources. These resources varied in how easily their function is reconfigured to match the task's demand (flexibility) and how independent they are from maintenance by the user (stability). The machines were built to solve arithmetic problems, which featured both volatile elements (values change constantly from trial to trial) and static elements (values do not change). A time- and work-efficient strategy is to allocate flexible resources to serve the more volatile elements and stable resources to the static elements.

With this task in mind, the general question of system building is divided into several more concrete questions. First, what strategy do participants adopt in their allocation of flexible and stable resources? Motivated by the common generalization that people seek strategies that minimize effort, whether physical or cognitive (Walton et al., 2006; Kool et al., 2010), **Hypothesis 1** predicted that participants would adapt their systems to match the demands of the task by assigning flexible resources to volatile problem segments and stable resources to static ones. Additionally, Hypothesis 1 also predicted that participants will respond to changes in the problem dynamics, such as when a hitherto static problem segment becomes suddenly volatile.

Second, although this experimental task is tightly controlled with respect to its resource tradeoff, this tradeoff could be upset by uncontrolled variation in extraneous factors in the participant's environment, such as variation in the ergonomics of their hardware and in their degree of engagement in the task. **Hypothesis 2** predicted that variation in these two factors will influence strategy selection.

In developing an environment comprising both technique learning and tool building, there are plenty of other questions to explore outside these hypotheses. Study 1 also examined participants' self-reports of other strategies, to include other resource allocation strategies as well as strategies at the varying levels of analysis. This included the lower level of sensory, motor, and cognitive operations, as well as the meta level of action where a person weighs the cost trade-off of seeking new strategies versus exploiting a current one. Additionally, to understand what drives optimal strategy, this study probes the explicit strategy-relevant knowledge that participants glean during the task.

Study 2 asks a new round of questions pertaining to distributed cognition and the notion of cognitive offloading. The task is configured such that participants can conduct part of the computation mentally (i.e., cognitive *onloading*), thereby reducing the size and maintenance demand of the machine. Intuitively, onloading should be more beneficial when the mental computation is easier (vs. harder) and when use of the external representation is more demanding (vs. less). Again assuming a tendency for minimizing effort, **Hypothesis 3** predicted that participants would be more likely to onload when mental computation easier, compared to when it is more difficult. As in Experiment 1, Experiment 2 also examines the responsiveness to changes in the problem dynamics, namely when the cost of onloading decreases or increases. Finally, as in Study 1, Study 2 tests the influence of an additional extraneous variable—math anxiety—on participant's decisions on onload the math problem versus offloading it to the machines they build.

Machine Building Task

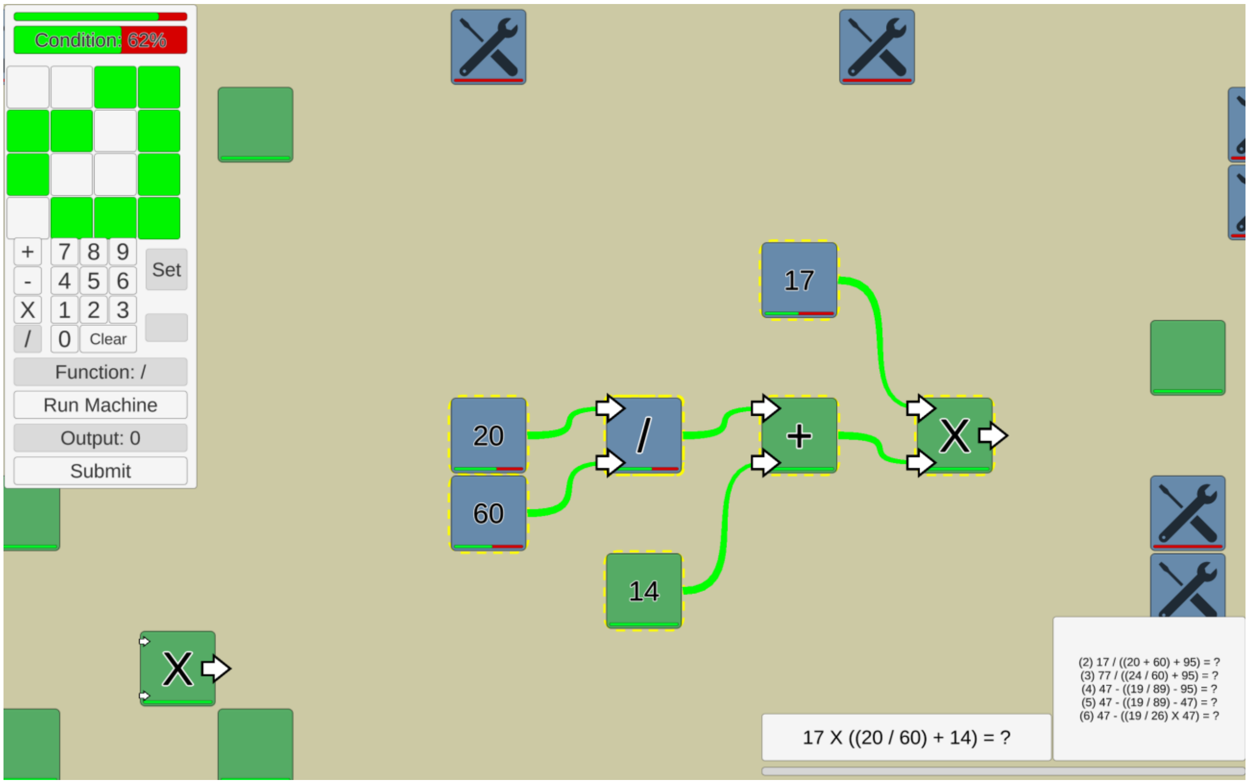
The machine building task (Figure 1) imitates basic visual programming in which two-dimensional blocks, representing number and arithmetic operations, are connected to one another in a node-link manner to form larger-scale algorithms. The task begins by displaying an array of empty blocks distributed across a two-dimensional space. The participant's goal is to configure and connect these blocks to build a calculator and run the prompted arithmetic computations on the bottom right of the screen. Beside the prompted problem, there is a preview of upcoming problems in smaller text.

A participant can pan their view over the space by clicking and dragging on the background. They can also move blocks by clicking and dragging them, and they can configure a block by clicking on it and interfacing with an inspector panel on the left of the screen. From this panel, a block can be configured as a number (ranging from 0 to 999) or as an operator (add, subtract, multiply, divide). Number blocks have a single output arrow that, by clicking and dragging, can connect to the input arrow of an operator block. Operator blocks have two input arrows and one output arrow. They can receive input from both number blocks and other operator blocks. For a given trial, blocks must be connected to one another to mirror the algebraic expression with its precise order of operations. Once the machine is assembled and all blocks are repaired, the participant selects any one of the blocks and then selects *Run Machine* in the inspector panel. An animation is run showing the steps of the computation. To submit the answer, the participant then selects the final block of the computation and selects *Submit* in the inspector panel, and a feedback message is presented. The trial only ends when a correct answer is given.

Blocks also demand maintenance in the form of a 4x4 array of tiles in the inspector panel. These tiles represent a block's condition, and certain events may cause tiles to switch to a white color signifying decay. If more than 8 tiles are decayed, the block will be become "broken" (indicated by a wrench icon) and will prevent any connected machine from completing a computation. Tiles are repaired by clicking on them, in which case their green color is restored. A block's maintenance condition is displayed in both the inspector panel and as a bar near the bottom of the block itself, with the green segment corresponding to the repair state of the block.

Flexibility and stability are implemented as blue blocks and green blocks, respectively. Flexible blocks can have their function switched at no cost, but they decay by two tiles every 15 seconds. Stable blocks do not decay over time, but switching their function causes a 12-tile decay.

Figure 1. Screenshot of the Machine Building Task (MBT). The configuration of connected blocks in the center corresponds to the arithmetic formula at the bottom of the screen.



Altogether, rather than providing participants with a hard-coded task environment, the MBT provides a hard-coded niche in which tools are built. In this way, the MBT demands strategy for both technique learning, tool selection and acquisition, tool customization, and system organization. For instance, at the level of motor sequences, participants might seek to optimize the sequence of actions in the most efficient way, such as by carrying out all actions for a given block (i.e., reconfiguration and repair) before moving on to another block. With respect to tool acquisition, they might selectively seek out one block type over another. With respect to customization, they might configure a block's spatial placement to match with the spatial arrangement of segments in the problem text. With respect to system building, they might evaluate a time cost tradeoff between exploring new system configurations versus exploiting their present one. There is also potential for interaction among these levels, such as the co-adaptive behavior in which routines and customization co-adapt (1990).

Developing the problem set. An advantage of using arithmetic problems is that because individual numbers and operators are discrete, problem segments can be manipulated independently from one another, allowing certain problem segments to change from trial to trial (volatile) while others remained the same (static). To build a problem set, a procedural generation script was developed that automatically creates arithmetic problems with a desired number of segments set so specific volatilities. The generator works by representing the problem as a network structure. It first instantiates an operator node, then queries parts of the problem with unfilled connections, and then adds a problem segment to that unfilled connection. If the number of operators instantiated so far is less than the target number, a new operator is instantiated at the selected input/output connection. Otherwise, a number node is added with a randomly selected number. This process continues until the target quantity of operators and numbers is reached. In this way a formula is generated with a precise order of operations. To create a subsequent trial, the existing formula is transformed by selecting specific nodes to alter. First, the algorithm selects nodes to be either high- or low-variety, based on input parameters. For high-variety nodes, if the node is an operator, the operator identity is transformed into a different operator (e.g., addition to subtraction). For high-variety number nodes, the number is changed to another random number (e.g., 25 to 54). For low-variety nodes, the identity never changes. In this way, each trial is an evolution of the previous trial. When trials for the first block have been generated, the algorithm creates a second block by re-creating the original problem and then evolving the problem using opposite configuration of variety conditions; high-variety nodes become low-variety and vice versa.

Additionally, checks are run to ensure that the number inputs are feasible to the user. To ensure participants used the machine blocks, it was also necessary to prevent users from simply substituting parts of the machine's computation with mental arithmetic. For instance, if a problem called for *2 + 2* in its formula, a participant might supply a *4* block instead of the three blocks that constitute the computation. To prevent this substitution, the result of each computational step was checked to ensure that it was a floating point value, which effectively prevents a substitution strategy (number blocks can only assume integer values). If the result was an integer value, a new number was randomly assigned and checked.

Ultimately, the problem set comprised 42 problems, each consisting of four numbers and three operators and bearing the same order of operations. The problems were also divided into two blocks. In one block, two of the numbers and two of the operators were volatile while the others were static. In the second block, the volatility assignments were reversed (one volatile operator, two volatile numbers).

Implementing stability-flexibility tradeoff. The goal of having flexible versus stable blocks is to establish a trade-off between the resources that are used, but this trade-off requires fine-tuning so that the advantages and disadvantages of the two blocks are comparable. For instance, if the penalty of using a stable block on a volatile problem segment is miniscule, then there may not be a true tradeoff between flexible and stable blocks. For this reason, a performance simulation was developed to test different parameterizations of the two blocks' repair demands and arrive at parameters that ensure the presence of a true tradeoff between the blocks.

The performance simulation was developed based on the assumption of an ideal performer. A given trial of the MBT is broken down into the essential steps of selecting blocks, changing their function, repairing them, and running the computation. An additional baseline time cost was included to cover additional steps, such as the time necessary to read and respond to the new trial prompt. For each trial, the simulation iterates through these steps, incrementing a time counter according to their individual time costs. As the simulated time accumulates, the simulation also determines how many tiles are expected to decay in each of the machine's blocks, taking into account the block type as well. When a block decays to breaking point, the steps necessary to repair a block (block selection and tile repair) are also added to the simulation's queue of operations.

For each run of the simulation, the time cost of these steps was an interpolation between an estimated minimum time and an estimated maximum time needed for that step to be completed. Thirty runs of the simulation were conducted, with each run assuming an interpolation point from 0 to 1. For instance, the cost of repairing a tile was 0.2 seconds (the minimum estimate) for the first run, 0.29 seconds for the 10^{th} run, and 0.5 seconds (the maximum estimate) for the 30^{th} run. In this way, the full simulation tested strategy performance across a wide range of potential performance speeds.

A single run of the simulation takes specific parameters regarding (1) the maintenance costs of flexible and stable blocks, (2) how many segments of a problem change per trial, (3) the resource-selection strategy (all-stable, all-flexible, problem-matching), and (4) the general speed of the participant in completing the task-intrinsic steps. The goal was to adjust (1) and (2) such that a resource trade-off emerged favoring a hybrid strategy (3), and that the advantage of this strategy was robust to a variation in the participant's speed (4).

Ultimately, flexible blocks were set to a temporal decay of 2 blocks every 15 seconds, and stable blocks were set to a decay of 12 blocks every time its function was changed, its number was changed, or its output formed a new connection. Figure 2 shows the differences in performance (time and repair cost) as a function of both the participant's general speed and the strategy selected. Critically, the problem matching strategy shows the minimum demand on repair and time compared to the other strategies.

Figure 2. Simulation's tile repairs as a function of strategy and speed.

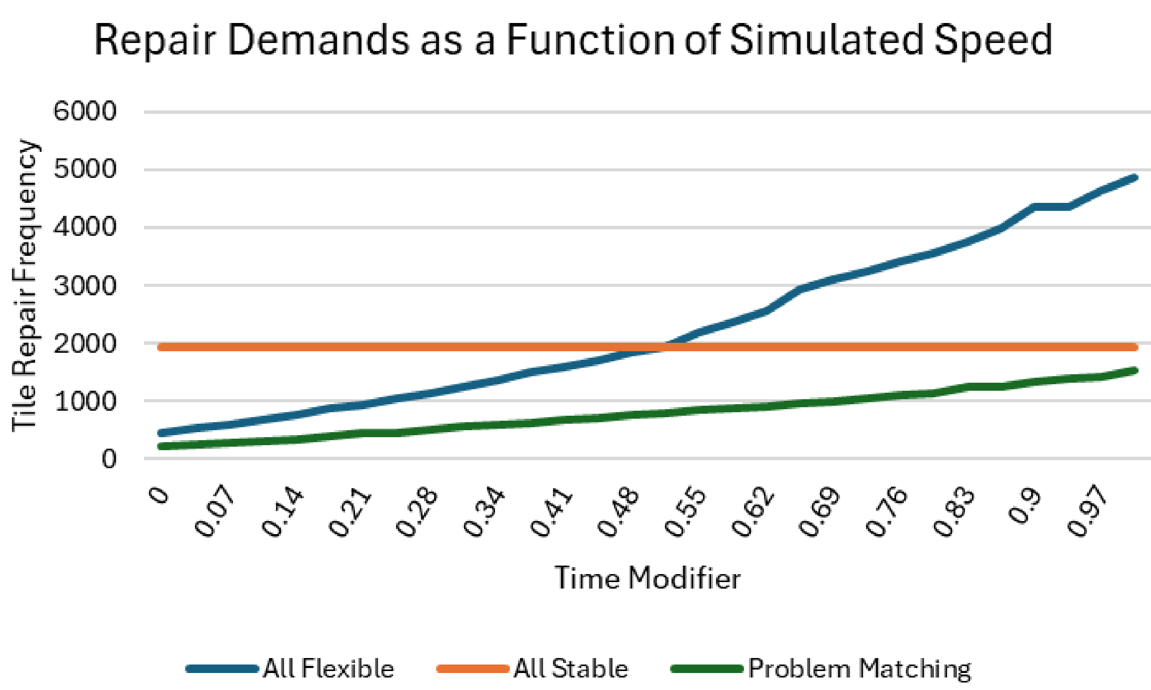
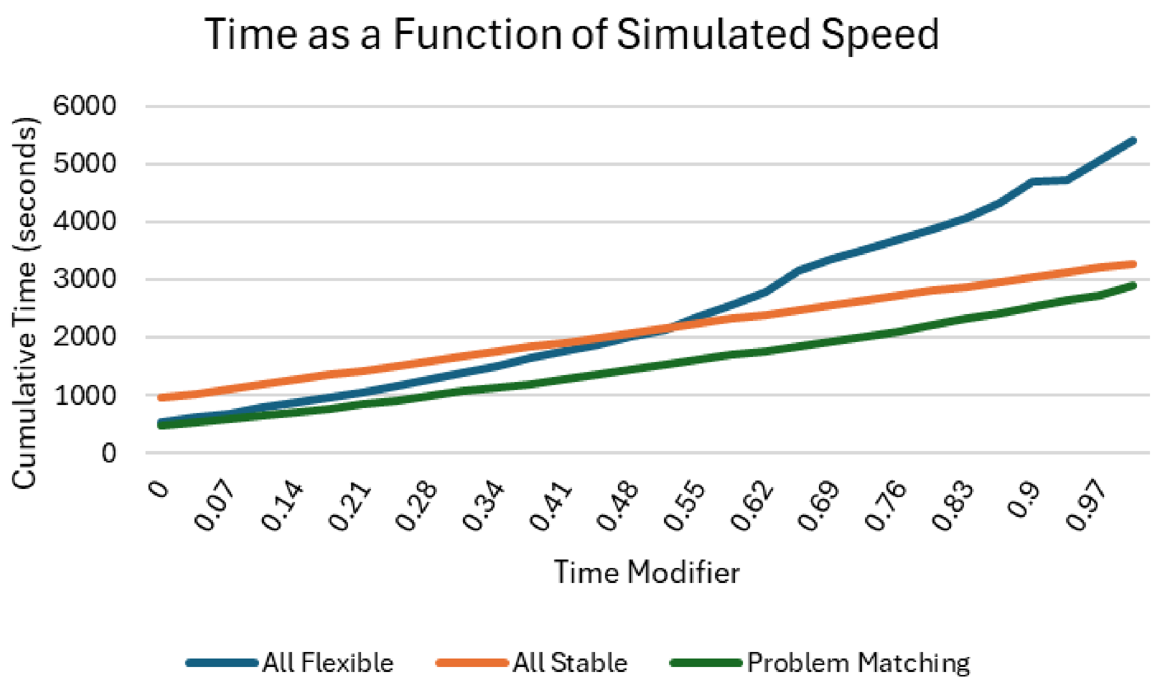


Figure 3. Simulation's cumulative time as a function of strategy and speed.



MBT Tutorial Video. To introduce the MBT to participants, a video was run explaining the task and the behavior of the blocks. The video explained the goal of the task, how to manipulate and connect computation blocks, and how to configure them using the inspector panel on the left of their screen. It also visually demonstrated the process of configuring, connecting, and running computations on blocks. Regarding the distinction between flexible and stable blocks, the differences between green and blue blocks with respect to maintenance cost were not mentioned, but participants were told that "different block types have different ways of behaving, but ultimately any block type can be assigned to any function you choose." During video demonstrations, machines possessed numbers and operators of both colors to signify this.

Cursor Movement Task

The cursor movement task is similar to the button task in Gray and Boehm-Davis (2000) and was designed to measure the latency at which participants moved their cursor to target a location on the screen. The goal of this measure is to account for variation in the participant's control over their cursor, which could depend on motor skill and the ergonomics of their remote workspace. For each trial of this task, a black square appeared in the center of a blank white screen, and upon clicking the square it disappeared. After a brief delay (random value ranging from 0.5 to 1 second), a black square then appeared in the periphery at one of eight possible directions from the center, and at one of three different distances, from the center (18 configurations). Upon clicking the peripheral square, the square disappeared and the central square instantly reappeared. The time between the onset of the peripheral target and clicking the peripheral target was measured for each trial.

Figure 4. Positions of targets in the cursor movement task.

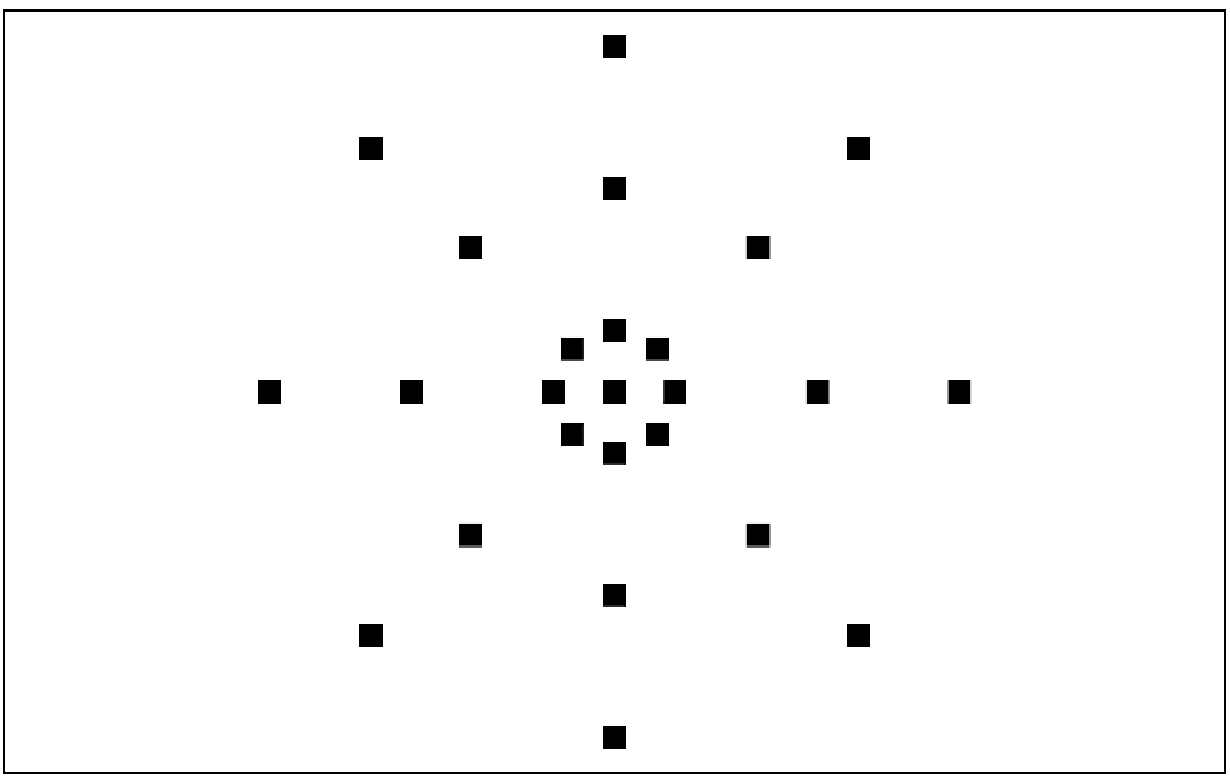
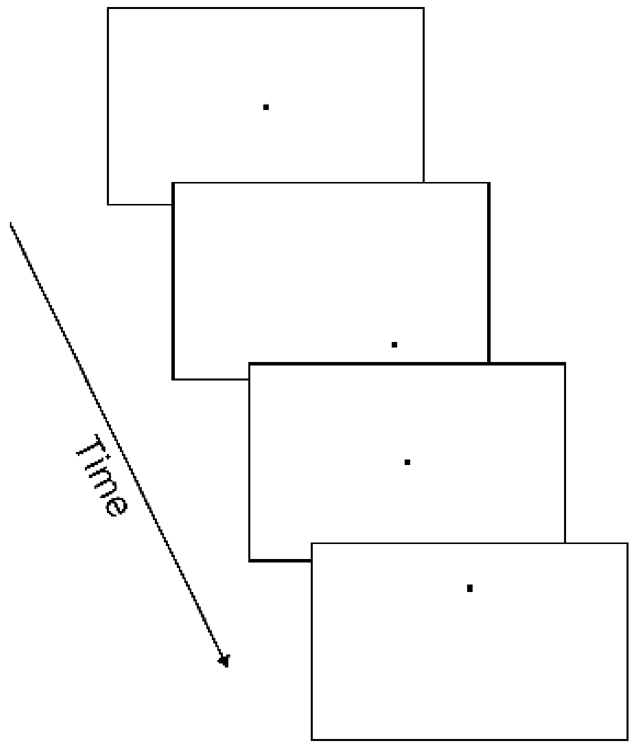


Figure 5. Example sequence of trials in the Cursor Movement Task



Study 1

Method

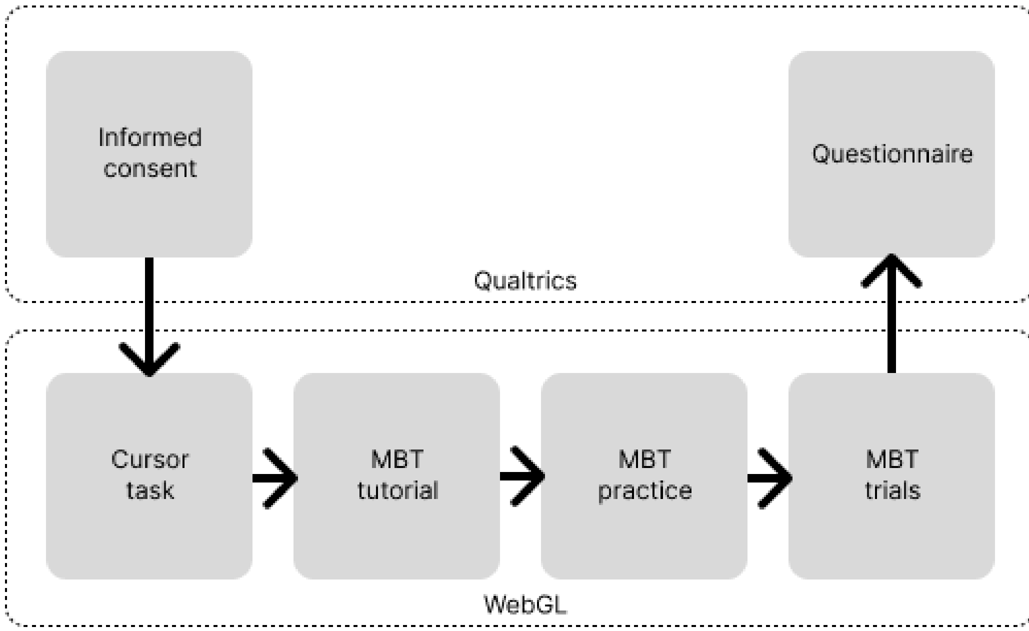
Participants

Recruitment was conducted over the Fall 2024 and Spring 2025 semesters until obtaining valid data for 15 participants in each of the two counterbalance conditions. Recruitment criteria included normal vision with no color vision impairment and access to working audio, mouse, and keyboard. Participants ($N = 44$) were recruited from the University's Sona research participant pool and from individual undergraduate classes and were given course credit for their participation. Of these, 35 participants completed the study. Of those who did not complete the study, all completed the initial meeting, but three later reported a technical problem preventing them from completing the study, and the other six did not return to the Qualtrics form to upload data or complete the remaining questions. Of the completed participants, one participant reported switching to a trackpad and so was excluded from analysis. Data from another four revealed repeat attempts. Cursory inspection of their data suggested significant performance differences in errors, so to avoid potential confounds of task experience or user frustration, those four were excluded from analysis. Only the remaining sample (N = 30, 14 females, M_{age} = 19.13 years, SD = 1.50) was included for data analysis.

Procedure

Figure 4 shows a summary of the procedure. Participants met the experimenter over Zoom for an orientation meeting, comprising review of the consent form and preliminary instructions on the completion of the Qualtrics survey and self-guided WebGL program. The experimenter verbally confirmed that the participants had a mouse, then gave them a link to the Qualtrics form, which began with the consent form. After consent, the participant viewed a page of instructions and followed a link to the WebGL program. Participants were told to open the program and keep both Qualtrics form and WebGL tabs open for the duration of the study. Participants were then given an ID number and a condition number, which they input into the WebGL program. The Zoom session was closed to allow the participants to complete the program on their own. Upon completing the program, they downloaded an encrypted data file from the WebGL program, uploaded this file to the Qualtrics form, and then completed questions on the form concerning their experience with the task (Appendix B). Specifically, they were asked what patterns they had noticed in the problem set, what differences they noticed in the green and blue blocks, what strategy they employed in choosing green or blue blocks, and what other strategies they used in building and maintaining their machines. They were then asked questions pertaining to demographics: age, sex, gender, college major, and year in college. Finally, they were asked if they had any final comments on the experiment.

Figure 6. Procedure for Studies 1 and 2.



WebGL Program

The WebGL program was built using Unity3D and hosted on GitHub. The WebGL platform allows a program to be run remotely in an embedded window on a web browser, requiring no installation on the user's computer. Upon opening the link to the program, the program displayed a notification that the embedded window must be maximized (i.e., full screen). This notification disappeared when the participant entered full screen mode, and it re-appeared if the participant left this mode later, ensuring that the task was always conducted with the total screen space.

On entering full screen mode, the main menu appeared comprising instructions, input fields for the participant ID and condition number, and a sequence of buttons corresponding to the participant's tasks (Appendix C). The buttons were selectively enabled and disabled in sequence to ensure that participants completed each task only once and in the right sequence.

First, the participant input session information (ID and condition) and pressed “Confirm Info,” enabling a button that begins the cursor movement task. On completing this task, a button was enabled corresponding to a tutorial video for the MBT. In the same manner, the participant completed 5 practice MBT trials, then 42 recorded MBT trials. Upon completion, a button appeared allowing the participant to download an encrypted data file storing performance data, and a password was displayed allowing the participant to continue to the next page of the Qualtrics form.

In addition to ensuring participants adhered to sequence, several measures were added to evaluate whether participants had made repeated attempts at the task and whether participants disengaged from the task. First, to identify repeat attempts, a counter was iterated each time the participant started the cursor movement task. Unlike the performance data, this count was stored as persistent data that lasted across browser sessions. Second, to assess lapses in activity, the program logged any instance in which both mouse and keyboard input had lapsed for at least 30 seconds. Third, to record when participants actively disengaged from the program, such as by minimizing the browser window, the program logged events in which the participant exited and restored the program’s full screen mode, and it logged events in which the program ceased being the system’s focus; this occurs when user’s input switches to other programs on the computer.

Results and Discussion

Flexible Resource Allocation

Hypothesis 1 predicted that in their design of their systems, participants would seek to minimize effort by adapting their resource allocation to the task demands. To test this, the fifth from the last trial of each block was sampled for each participant. This trial was selected as a point at which the trial block was nearly complete but a preview of the next trial block had not appeared, ruling out any possibility of participants changing their machine in anticipation of a change in problem volatility. Each trial's problem comprises multiple segments corresponding to multiple computation blocks, each of which is either flexible or stable. Given the nested nature of these data, a multi-level logistic regression was conducted with the outcome variable the assignment of resource type (0 = stable, 1 = flexible) and the predictor variables including problem segment volatility (0 = static, 1 = dynamic), and participant. Using R Studio, models were developed such that predictor variables were added iteratively with each new model to determine the influence of a given variable on the predictive strength of the new model.

Table 1 shows the results of each model. The first of these is a null model comprising only a random intercept for the participant. For the next model, volatility was added as a fixed effect to determine its general influence on the selection of a flexible resource. This significantly improved the model fit as shown by its reduction in Akaike Information Criterion value (ΔAIC = 10.9), with participants 3.09 times more likely to assign a flexible resource to a dynamic problem segment. Comparing the models revealed a significant difference ($\chi^2 = 12.96$, $p < .001$). For the third model , volatility was added as a random intercept to assess whether the influence of a problem segment's volatility varied among participants. This resulted in a further improvement to the model fit (ΔAIC = -4.7), although the fixed effect of volatility was no longer significant. Comparing the interaction model with the volatility model also revealed a significant difference ($\chi^2 = 8.64$, $p = .01$). This indicates that while the volatility of the problem segment influenced participants' resource allocation, the degree of influence also varied significantly among participants. Therefore, Hypothesis 1 is only supported with respect to some participants' behavior, while failing to predict the strategies of others.

Table 1. Logistic Models Predicting Selection of Flexible Block

Predictor	Null Model	Volatility Model	Volatility x Sub Model
(Intercept)	0.30*** [0.16, 0.55]	0.15*** [0.07, 0.33]	0.20*** [0.11, 0.37]
Volatility	-	3.09*** [1.61, 5.93]	1.88 [0.65, 5.41]
AIC	267.50	256.60	251.90
BIC	274.50	267.00	269.30

Note: *** p < .001. Estimates are ORs with confidence intervals.

Self-reported Knowledge and Strategies

The final regression model showed heterogeneity in how participants adapted to the flexibility-stability tradeoff. To better understand this heterogeneity, responses on the Qualtrics form questions were coded to classify the strategies and knowledge participants inferred during the MBT. For each participant, the self-reported block selection strategy was grouped into one of several categories (Table 2). Consistent with earlier findings, eight of the 30 participants reported opting for the problem matching strategy, and comparable numbers of participants reported prioritizing stable blocks or having no strategy altogether.

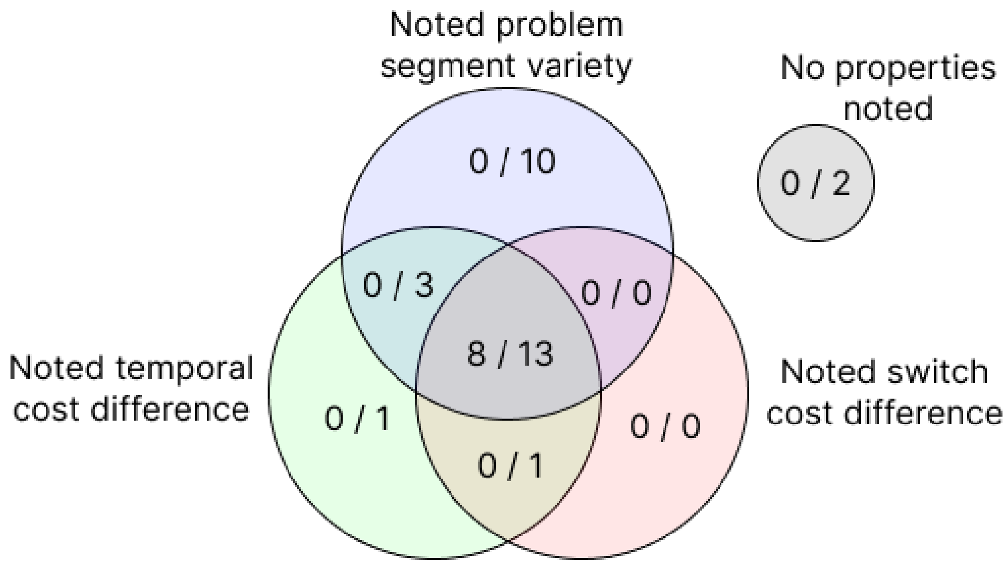
To measure task-relevant knowledge, participants were dummy coded on whether they noted (1) that problem segments varied in volatility, (2) that segments changed in their volatility midway through the trials, (3) that blocks differed in modification-induced decay, and (4) that blocks differed in temporal decay. Figure 5 shows the proportion of participants who reported committing to the resource-matching strategy as a function of the task properties they reported.

Of the 13 participants who recognized all strategy-relevant task features, eight reported the problem matching strategy. Critically, no other participants opted for the strategy, suggesting that participants had developed this strategy through explicit knowledge gained from observations of the block behavior and problem set.

Table 2. Self-reported strategies on the allocation of flexible and stable blocks.

Name	**Description**	**Participants**
Match problem	Match block types to problem segment volatility.	8
All stable	Allocate stable blocks to all problem segments.	7
All flexible	Allocate flexible blocks to all problem segments.	1
Match function	Match block types to block functions	4
Precedent	Allocate blocks based on the tutorial video	1
None	Mentioned having no strategy, or no strategy mentioned pertaining to block type.	9

Figure 7. Problem-matching strategy as a function of knowledge. Venn diagram indicates the number of participants (denominator) reporting task-relevant properties. Numerator indicates the number of participants in that bin who also reported using the optimal hybrid strategy.



Strategy and Performance

In principle, the problem matching strategy should minimize the repair demand and therefore the time on the task as well. Yet surprisingly few people opted for this strategy, so was this strategy really optimal? To verify this, an alignment score was calculated for each trial as the proportion of blocks in the machine that were consistent with this strategy. Two linear mixed effects models predicting trial repairs and trial duration were run with alignment as a fixed effect and participant as a random effect. Trials in which a disengagement occurred (discussed later) were excluded. The repairs model shows that machine alignment significantly predicted the number of repairs per trial (Table 3); machines demanded fewer repairs when they more closely aligned with the problem matching strategy. Likewise, the trial duration model also revealed that alignment significantly predicted shorter trial time.

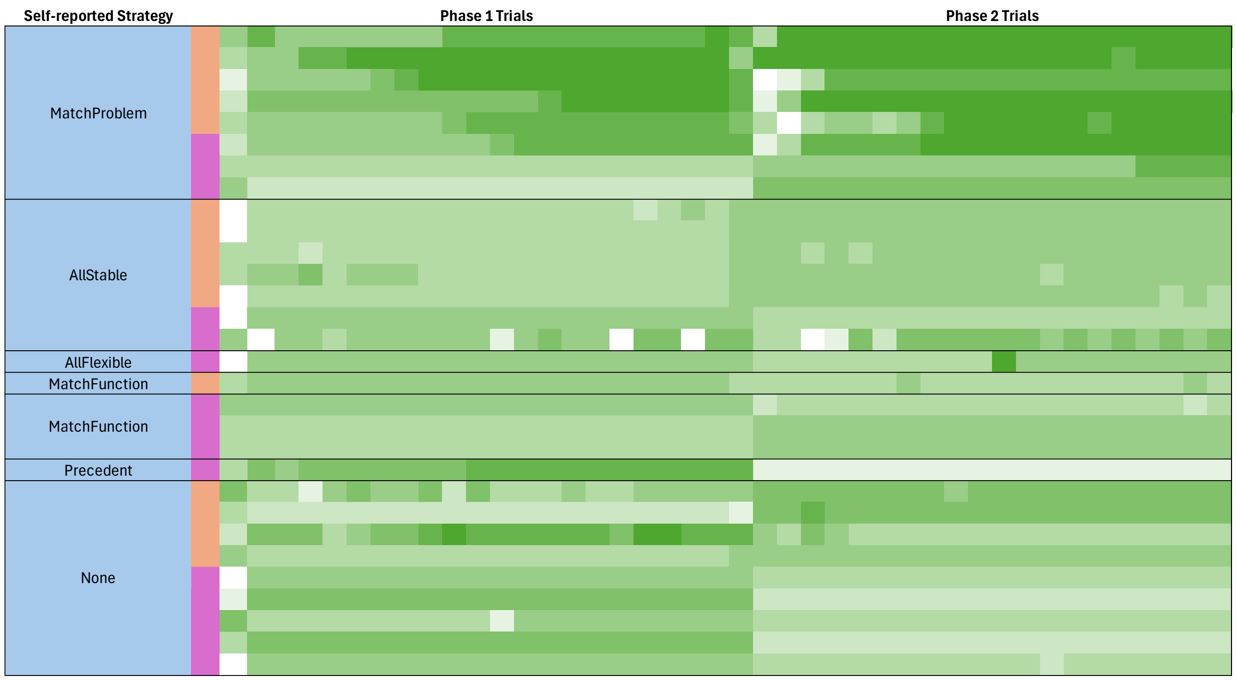
Table 3. Models predicting trial repairs and trial time for Study 1

Criterion	**Predictor**	***b***	***SE***	**95% *CI***	**β**	***p***
Repairs	(Intercept)	4.76	0.09	[4.57, 4.94]		<0.001
	Alignment	-2.14	0.11	[-2.37, -1.91]	-0.50	<0.001
Trial duration	(Intercept)	4.23	0.06	[4.11, 4.35]		<0.001
	Alignment	-1.04	0.07	[-1.17, -0.91]	-0.45	<0.001

Another interesting question is how participants' self-reported block selections aligned with the problem matching strategy. Figure 6 shows the degree of resource alignment by trial and by participant, sorted according to the self-reported strategy. The figure shows that most of those who reported the problem-matching strategy attained complete or near-complete alignment, and that this alignment persisted. Remarkably, at the onset of the second trial block (when the segments alternated their volatility assignments) the machines lost alignment but were quickly re-aligned to match the problem volatility. Additionally, no participants reporting alternative strategies showed persistent alignment, meaning that participants opting for these alternative strategies did not incidentally optimize their machine to the level of the optimal strategy.

Perhaps a more interesting question is why participants opted for alternative strategies in the first place, especially those who understood the essential task properties. For instance, a surprising number of participants reported a different matching strategy in which they allocated color of block according to the block's function. This is despite efforts in the experiment design to mitigate such an inference: trials feature both static and dynamic instances of both number segments and operator segments, and tutorial instructions also gave examples of both.

Figure 8. Machine alignment to problem matching strategy. Each cell shows the degree of alignment between the blocks used in the solving computation and the blocks that a problem matching strategy would employ. Each row shows the sequence of trials in temporal order from left to right. Greater intensity of green indicates greater alignment. Orange = condition 1, purple = condition 2.



Levels of Strategy Reported

For an understanding of participants' general strategies, each participant was dummy coded on whether they commented on a strategy of a specific level of behavior; these levels included (1) sensory, motor, and cognitive operations, (2) block selection, (3) other machine configuration strategies, and (4) meta-strategies concerning the decision to explore alternative routines vs. persisting in the present one. Table 4 provides an overview of how frequently participants commented on levels of strategy. Note that in the survey, the only level of these that was explicitly solicited was resource selection. However, participants also volunteered a comparable number of comments on action sequence strategies and strategies pertaining to the machine's configuration. The significance of this is that when engaging with this building task, the question of optimization does not pertain strictly to the choice of green versus blue blocks but to other levels of analysis as well.

Disengagement and Motor Skill

Participant logs also recorded data on disengagement, including instances in which the participant was idle for more than 30 seconds, exited the program's full screen mode (which effectively disables the program's interface), or switched the input focus away from the program. To better understand disengagement, the frequencies and cumulative durations of these events were tallied for each task within the program, as well as for interim periods occurring between the end of one task and the beginning of the next.

Appendix D shows the frequency and cumulative duration of each type of disengagement, organized according to the segment of the WebGL program. Disengagements were relatively infrequent during the cursor task and MBT. Only one participant showed disengagements during the cursor task (exited full screen for 3.2 seconds). For the main MBT trials, seven showed some manner of disengagement, with their cumulative durations ranging from 0.9 to 238 seconds. Disengagements are markedly more frequent in the interim phase, suggesting that participants tended to take breaks between but not during tasks. Idle periods in the video phase are frequent but unremarkable, as this segment did not demand any user input.

Table 4. Self-reported strategies at four levels

Level	Description	Example	Count
Action sequences	Coordinating sequence of perceptual, motor, and cognitive actions.	"Instead of waiting for the condition to run out I would just maintain them as close to 100% so it would be less work at one time."	19
Resource selection	Selecting blue vs. green blocks.	"I used only green blocks because all the problems would only change 3 variables every new question, so having all the blocks lose durability overtime seemed inefficient."	21
Configuration	Other strategies on machine organization (e.g., spatial placement of blocks)	"I visually had my machine laid out in an easy to follow pattern and could reference the same blocks when needing to change elements."	13
Meta-strategy	Deciding on what strategy to learn or whether to spend time learning a strategy.	"[…] I learned the behaviors of the different blocks as time went on but I did not want to use excess time to set up a strategy with the information I learned."	2

Do Disengagement and Motor Skill Influence Strategy (Hypothesis 2)? Because the flexibility-stability tradeoff partly depends on time, and because the task was conducted remotely in an uncontrolled setting of the participant's choice, it is possible that temporal hindrances extraneous to the task could offset this resource tradeoff to such a degree that a stability-focused strategy became favorable to problem matching strategy. For instance, if a participant was distracted for 60 seconds, this would be long enough to decay flexible blocks by half their tiles. Likewise, if a participant's mouse configuration was awkward, the added hindrance could have imposed a systemic delay that offset the tradeoff.

To test the association between disengagement and an all-stable strategy, disengagement was dummy coded to group participants showing any idle time (which would necessarily exceed 30 seconds). Of the five participants in this category, four reported using an all-stable strategy. Cumulative time spent on cursor task was also included as a proxy for motor skill. A multiple regression was conducted to determine whether disengagement and cursor task time predicted the all-stable strategy (Table 5). Compared to a null model, the overall model was significant ($\chi^2 = 9.53$, $p = .009$). However, only disengagement significantly predicted the all-stable strategy ($p = .008$). Time on the cursor task was not a significant predictor.

In short, only disengagement successfully predicted a strategy prioritizing stable resources, lending partial support for Hypothesis 2. However, it also bears mentioning that with only five participants who went idle, and odds ratio of this test may be inflated. Furthermore, this finding alone does not reveal the causal mechanism—whether disengagement led to the stability strategy or a stability strategy made participants feel more inclined to disengage. Performance logs reveal that the all-stable configuration was adopted prior to disengagement, suggesting the latter explanation.

Table 5. All-stable strategy as a function of disengagement and cursor task time

Predictor	β	*SE*	*z*	*p*	*OR*	95% *CI* for *OR*
(Intercept)	–.33	3.11	–.11	.915	.72	[.0008, 287.50]
Disengagement	3.47	1.31	2.66	**.008**	32.26	[3.32, 812.15]
CursorTaskTime	–.05	.09	–.53	.594	.95	[.79, 1.14]

Study 1 Summary

The goal of Study 1 was to examine people's behavior in a context in which they do not merely work with tools provided but build and modify their own usable system from a variety of smaller-scale resources. The central question was whether their resource selection adapted to the constraints of the task, in line with a notion that participants would seek to minimize time and work on the task. However, participants proved heterogeneous in the resource selection strategies they adopted. Further examination of self-reported strategy selection was not isolated to the question of resource selection but occurred at multiple scales, especially at the level at which action sequences were planned.

Study 2

Study 1 was designed such that participants could not use mental arithmetic and their solutions relied completely on the machine's computations. However, if participants had the opportunity to calculate solutions in the head, would they do so? This notion of preferring in-the-head computations puts an interesting spin on ecological approaches to cognition, as the focus tends to be the reverse process. *Cognitive offloading* refers to the use of external resources such as a calculator to alleviate cognitive demands (Risko & Gilbert, 2016). Offloading includes behaviors such as using fingers to count, but it also includes the use of devices designed to support, supplement, or supplant cognitive processes, variously referred to as cognitive artifacts (Norman, 1991) and cognitive aids (McLaughlin & Byrne, 2020). This can in part be attributed to a general preference toward minimizing cognitive load (Kool et al., 2010). However, offloading is not always beneficial or preferred. For instance, people are found to avoid offloading when the external representation is unreliable (Weis & Wiese, 2019) or has a high temporal cost to access (Gray et al., 2006), and there is also evidence that strategy selection not always optimal (Dunn & Risko, 2016).

Furthermore, it is arguably incomplete to speak only of offloading when the allocation decision is arguably bidirectional. If mental processes are recruited to alleviate physical burden (say, of manipulating and maintaining a calculator), then such behavior might well be called cognitive *onloading*. The point is not just a matter of principle. For instance, when students learn to use a physical abacus to make complex arithmetic calculations, they automatize the routines for manipulating the artifact, and they may develop such expertise that they can mentally simulate the artifact and its manipulations as a surrogate to the artifact itself. In some cases, the "mental abacus" has been found to be advantageous in that it eliminates the time and motor cost of manual manipulation (Hatano et al., 1977).

The goal of this experiment was add the cognitive allocation decision as an additional degree of freedom to participants' decision making, and in this new context determine whether findings from Study 1 replicate and then examine cognition allocation strategies. Assuming again the people seek to minimize cognitive and motor effort, **Hypothesis 3** states that commitment to mental arithmetic will depend on the differential costs of the onloading versus offloading options. Therefore, between two conditions of different difficulties, offloading is expected to occur more often in the more difficult condition. By the same token, the hypothesis also predicts that people should respond to changes in the cost balance. Offloading should occur if the computation becomes suddenly difficult, and onloading if suddenly easy. However, another possibility is that people perseverate in their strategies after settling on a usable system. A classic finding of problem solving is that people are susceptible to *Einstellung,* in which they fail to recognize a new favorable strategy and instead perseverate in one that is no longer optimal (Bilalić et al., 2010; Luchins, 1942). In revisiting Luchins' original data, Binz and Schulz (2023) explained Einstellung not as a maladaptive behavior but as one that accounts for the cost of continual strategy exploration.

As Study 1 also examined the influence of extraneous factors on strategy selection, for Study 2 math anxiety was selected as a factor that may influence cognition allocation. Math anxiety has been found to predict lower performance on mental arithmetic (Ashcraft & Faust, 1992). So as an additional prediction of the effort minimizing hypothesis, the expectation is that higher reports of math anxiety will predict greater offloading strategy.

Method

Participants

Participants were recruited over the 2025 Spring semester from the university's Sona research participant pool in exchange for credit in an introductory psychology course. To ensure that the two experimental conditions had a comparable distribution of math anxiety scores, the target sample size was doubled ($N = 60$, 12 females, $M_{\text{age}} = 19.6$, $SD = 1.81$), and recruitment continued until reaching that count. In total, 85 participants were recruited. Of those, 15 failed to complete the study; four of these reported being unable to complete the program due to technical errors, and the remaining 11 did not follow up on the Qualtrics form. Of the 70 who completed the study, nine were excluded on account of having repeated attempts, and one was excluded for noncompliant answers on the Qualtrics survey.

MBT Revisions

The same procedure as Study 1 was used again in Study 2 but with several revisions. In the tutorial video, additional instruction was added:

> "Notice that to complete a problem, you only need to have a block with the solution stored as an output. This means that it is not always necessary to have a block for each number and operator, just as long as you have a block that contains the correct final answer. For instance, suppose you see this problem. The first operation is very simple: 2 plus 2. Instead of setting up two number blocks and an operator block, you can set up a single 4 block followed by the rest of the computation. If you are choosing between building a complete machine and taking this kind of shortcut, choose the option that you feel is faster and more accurate. However, you must not use any additional aids, like calculators or pencil and paper to make your calculation."

The algorithm generating the problem set was modified so that the first computation in the problem would be feasible using mental arithmetic. To ensure this, the algorithm only created computation that resulted in integer solutions. Furthermore, to manipulate the difficulty of the mental arithmetic segment, the two trial blocks were designed to differ in the difficulty in the numbers in the computation. For the easier trial block, the mental computation step involved one-digit addition and subtraction (e.g., 4 + 3), whereas the other block comprised multiple two-numbers (e.g., 57 + 14).

Additionally, the segments of the problem set were configured such that for both of the trial blocks, the mental computation step comprised two high-variety segments and one low-variety segment. This configuration entailed the same maintenance payoff for both conditions—specifically, the onloading strategy obviates the need to reconfigure the two blocks continually.

Additionally, in response to participant comments in Study 1, a learning aid was added to the MBT in the form of a tooltip-enabled graphic (Appendix E). When the tooltip was clicked, a graphic summary appeared showing the steps necessary to assemble and run a machine. To avoid encouraging any preference in block type, the graphic included numbers and operators in both colors.

Abbreviated Math Anxiety Scale (AMAS)

Between the questions pertaining to the task and the demographics questions, a math anxiety scale was added to the final questionnaire (Appendix E). The Abbreviated Math Anxiety Scale (AMAS) has been validated by Hopko et al. (2003) and comprises questions in which participants rate their degree of anxiety in specific situations demanding math skill, such as pop quizzes in a classroom.

Results and Discussion

Replication of Study 1

As in Study 1, the same multi-level logistic regression and qualitative coding was conducted to assess participants' resource allocation strategies. As before, the null model was significant ($p < .001$). Adding volatility as a fixed effect to the model improved its fit ($\Delta AIC = -19.2$), with volatility proving a significant predictor of block type allocation. The new model also proved significantly different from the null ($\chi^2 = 21.16$, $p < .001$). Use of a flexible block was 2.97 times more likely when the block was to have its value switched. Adding volatility as a random effect also made a significant improvement to fit (ΔAIC = -25.10), but once again the fixed effect of volatility lost significance. The interaction model proved significantly different from the fixed effect volatility model ($\chi2 = 29.12$, $p < .001$).

Table 6. Model comparison predicting flexible resource allocation for Study 2

Predictor	**Null Model**	**Volatility Model**	**Volatility x Sub Model**
(Intercept)	.38 [.25, .58]***	.23 [.14, .38]***	.20 [.11, .39]***
Volatility	-	2.97 [1.82, 4.86]***	2.86 [.94, 8.68]
AIC	459.50	440.30	415.20
BIC	467.40	452.20	435.00

Note: *** p < .001. Estimates are ORs with confidence intervals.

Cognitive Strategy

For each trial, the participant could choose whether to complete the first computational step externally, relying on a seven-block machine, or internally, thereby reducing the machine to five blocks. Mental math was thus operationalized as any trial with fewer than seven blocks recruited in the problem-solving computation. Figure 7 shows the proportion of trials involving mental math for each interaction of math complexity (simple vs. complex) and experiment condition (simple trials first vs. complex trials first).

To determine whether problem complexity and trial block order influenced participants' decisions between internally computing the intermediate answer or using a machine, a *t*-test was first run to verify that the AMAS score did not significantly differ between condition 1 (M = 22.17, SD = 6.21) and condition 2 (M = 21.57, SD = 7.27). This test showed no significant difference ($t = .34$, $df = 56.63$, $p = .73$). Additionally, a Shapiro-Wilks test revealed that the distribution of AMAS scores did not significantly differ from normal ($W = .97$, $p = .21$). A multi-level logistic regression was then used, beginning with a null model and adding difficulty followed by block order (Table 7). The null model proved significant ($p = .03$), indicating a general preference for mental math on the critical step. However, the addition of math complexity failed to reach significance ($p = .223$) or improve the model fit. Likewise, the addition of block order ($p = .50$) and its interaction term ($p = .82$) failed to reach significance. Finally, the AMAS score was added as a predictor variable to the existing logistic regression, resulting in the third model. AMAS score also proved to be non-significant ($p = .774$).

Altogether, problem difficulty, difficulty order, and AMAS score failed to predict whether participants resorted to cognitive offloading or onloading. This is surprising but might be explained by the strength of the manipulation between easy and difficult conditions. Presumably, for offloading might not have appeared advantageous if the anticipated cost of switching and maintaining two additional blocks failed to offset the anticipated cost of mental computation. The overall preference for mental math suggests this. Thus, the results are consistent with hypothesis 3, but not in a way that is especially informative.

Figure 9. Mental math as a function of experiment condition and difficulty

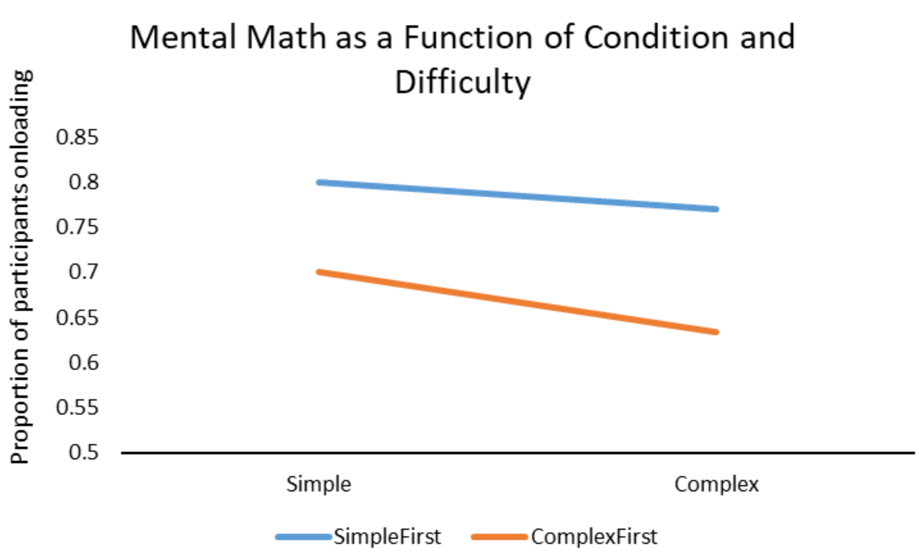


Table 7. Model results predicting mental math41

Predictor	Null Model	Model 1	Model 2	Model 3
(Intercept)	6.34 [1.19, 33.83]*	3.90 [.63, 24.23]	2.13 [.17, 26.45]	14.21 [0.04, 4566.36]
Difficulty	-	2.95 [.52, 16.74]	3.56 [.34, 36.85]	-
Difficulty order	-	-	3.53 [.09, 141.29]	-
Interaction	-	-	0.66 [.02, 22.67]	-
AMAS score	-	-	-	0.96 [.75, 1.24]
AIC	109.123699	110.036876	112.5269533	110.8
BIC	114.6986825	118.3993512	126.464412	119.2

Note: Predictor values refer to odds ratios and confidence intervals. * $p < 0.05$.

Strategy and Performance

How did block selection strategy and the onloading strategy predict repair efficiency and trial time? The same analysis from Study 1 was applied to predict these outcome variables, but with the addition of mental math as a predictor variable. Table 8 shows the results of these models. Higher degrees of alignment and mental both significantly predicted lower values. The significant time payoff of the onloading strategy may help explain why the manipulation of difficulty had no significant effect on the onloading strategy.

While not the focus of the replication, alignment scores, strategy self-report data, and disengagement data were calculated for participants as with Study 1, These can be found in Appendices G, H, and I.

Table 8. Models predicting trial repairs and trial time for Study 2

Criterion	Predictor	*b*	*SE*	95% *CI*	ß	*p*
Repairs	(Intercept)	5.29	0.12	[5.05, 5.52]	<0.001	<0.001
	Alignment	-3.25	0.12	[-3.49, -3.02]	-0.71	<0.001
	MentalMath	-0.56	0.09	[-0.73, -0.38]	-0.22	<0.001
Trial duration	(Intercept)	4.28	0.06	[4.15, 4.41]	<0.001	<0.001
	Alignment	-1.33	0.06	[-1.46, -1.20]	-0.61	<0.001
	MentalMath	-0.32	0.05	[-0.41, -0.22]	-0.26	<0.001

General Discussion

Adaptive use of external resources

Study 1 posed the question of how people adapted their tool systems to the demands of their task. The critical measure for adaptation was the degree to which they allocated flexible versus stable resources to volatile versus static parts of their system—the problem matching strategy. Prior simulations and performance data confirmed that this strategy reduced work and time. However, few participants actually adopted the strategy. Self-report data suggest that this failure could in part be attributed to variation in participants' awareness of the strategy-relevant properties of the task. After all, it was exclusively the participants who understood the block properties and the problem dynamics who configured their machines optimally.

However, the failure cannot be solely attributed to knowledge; many participants who did understood these critical elements of the task still opted for an alternate strategy. Of these participants, two stated that they did not want to spend additional time inferring a strategy. In this regard, participants are focused on a different optimization problem, which is one of balancing exploration and exploitation, or a search for a good answer versus investing the time to completing the task with the solution at hand (Cohen et al., 2007). Meanwhile, another had recognized strategy-relevant properties but opted for an all-stable strategy, mentioned that the temporal decay of the flexible blocks was stressful. Time pressure has a recognized influence on decision making (Philips-Wren & Adya, 2020). In assessing cognitive load, the NASA Task Load Index includes time pressure as an individual item (Hart, 2006). In this experiment, the time pressure of flexible blocks was an essential part of the resource tradeoff and was fine-tuned against the cost of stable blocks. Surprisingly, one participant commented that they enjoyed the sound of the repair tiles to an extent that they chose to avoid optimization (They had already aligned their machine to the problem, however.).

Aside from task-intrinsic features, both studies examined extraneous factors outside the control of the MBT environment. Study 1 found that participants who disengaged tended to opt for an all-stable strategy, which effectively alleviated time pressure. However, motor skill, as measured by the cursor movement task, did not predict a stabilization strategy.

Summing up, the question of whether people successfully optimize their configuration and workflow to minimize effort, whether in terms of time or effort, has a complicated answer. The results here this suggest that in the context of tool recruitment, customization, and system building, people should not be expected to optimize their artifacts and tools with respect to one particular line of performance, such as with the properties of specific materials. Furthermore, because people are engaged at optimization decisions at multiple scales, it is possible that one scale might be emphasized over another, although future research should more directly examine this question.

Internal versus external resources

Cognitive offloading is a common phrase referring to our reliance on the environment for much of our memory and reasoning, and it is a central piece to ecological approaches in psychology (Risko & Gilbert, 2016). Curiously, there has been seemingly no empirical interest in its conceptual counterpart *onloading*, which is just as coherent in principle. Study 2 introduced the decision between offloading and onloading as a new dimension to the task. Participants could rely entirely on the machine for computations as in Study 1, or they could complete the initial part of the computation mentally and manage a smaller machine. It was expected that this allocation decision would depend on the differential burdens the two options imposed on the participant, and that when problems became sufficiently difficult, cognitive demand of the onloading strategy would outweigh the physical demand of the offloading strategy. However, most participants opted for mental arithmetic regardless of the manipulation, and no significant strategy difference was found between simple problems and complex problems. Furthermore, participants' ratings on math anxiety also did not significantly predict any difference in strategy. The most plausible explanation for this is a failure to establish a sufficiently strong manipulation between the two conditions. The finding is not inconsistent with a minimal effort hypothesis, but it is not especially insightful either. For potential future studies examining this question, especially using the MBT, additional fine-tuning will be needed to establish parity between the offloading strategy and the onloading strategy.

Empirical implications

The study of perceptual-motor strategy, technology adoption, customization, and cognitive offloading all have precedents in empirical literature. However, these precedents tend to examine these facets of technology in piecemeal while treating others facets as hard constraints, reducing the user's degrees of freedom. By contrast, the task presented here provided a niche in which people are simultaneously developing techniques, adopting tools, and creating and modifying composite systems of these tools to complete a cognitive task. This broad activity of system building thus entailed strategy selection at multiple levels: the control and sequencing of sensory, motor, and cognitive actions, and the modification of one's external world through the recruitment and organization of artifact resources.

Future research should continue to examine the interplay between these two levels of strategy. However, as a limitation this environment falls short of the activity theory and distributed cognition approaches with respect to the social and cultural aspect of work. There are two appealing ways that future studies could explore this dimension. Implementing multiplayer in the MBT could provide an avenue for exploring how practices at the action level, artifact level, and social level co-evolve. Second, an emerging domain in anthropology and psychology is the study of cumulative technological culture, which examines the evolution of artifacts over generations (Osiurak & Reynaud, 2020).

Methodological implications

These studies also make a novel methodological contribution in using a flexible and remote testing method on a complex experimental task involving perception, motor control, and time-sensitive action. This is both a limitation and a strength. Of course, a natural concern of this method is whether participant disengagement could undermine a time-sensitive task. It is for that reason that these studies employed several ways of measuring engagement: querying previous attempts at the WebGL program, detecting idle times, detecting changes in the screen size of the program, and detecting instances when the program lost input focus. In doing so, this study leveraged this limitation into a potential a source of variety that could give a better insight into strategy selection.

Conclusion

In a seemingly straightforward task of balancing flexibility with stability, participants were not necessarily optimal, but they showed remarkable heterogeneity in the strategies they adopted. The focus of their strategies ranged from fine-grained motor and cognitive operations to the higher-level configuration of artifact systems. These findings highlight a need to examine decision making in environments where people are not merely users, but have a crafting hand in their work environment.

References

Ashby, W. R. (1991). Requisite Variety and Its Implications for the Control of Complex Systems. *Facets of Systems Science*, 405–417. https://doi.org/10.1007/978-1-4899-0718-9_28

Ashcraft, M. H., & Faust, M. W. (1994). Mathematics anxiety and mental arithmetic performance: An exploratory investigation. *Cognition & Emotion*, *8*(2), 97-125.

Ascher, M. (2008). The logical-numerical system of Inca quipus. *Annals of the History of Computing*, *5*(3), 268-278.

Bilalić, M., McLeod, P., & Gobet, F. (2010). The mechanism of the einstellung (set) effect: A pervasive source of cognitive bias. *Current Directions in Psychological Science*, *19*(2), 111–115. https://doi.org/10.1177/0963721410363571

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

Barricelli, B. R., Cassano, F., Fogli, D., & Piccinno, A. (2019). End-user development, end-user programming and end-user software engineering: A systematic mapping study. *Journal of Systems and Software*, *149*, 101-137.

Bødker, S., & Klokmose, C. N. (2011). The human–artifact model: An activity theoretical approach to artifact ecologies. *Human–computer interaction*, *26*(4), 315-371.

Burkolter, D., Weyers, B., Kluge, A., & Luther, W. (2014). Customization of user interfaces to reduce errors and enhance user acceptance. *Applied Ergonomics*, *45*(2), 346-353.

Cajori, F. (1993). *A history of mathematical notations* (Vol. 1). Courier Corporation.

Cohen, J. D., McClure, S. M., & Yu, A. J. (2007). Should I stay or should I go? How the human brain manages the trade-off between exploitation and exploration. *Philosophical Transactions of the Royal Society B: Biological Sciences*, *362*(1481), 933-942.

Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS quarterly*, 319-340.

Dreisbach, G., & Fröber, K. (2019). On how to be flexible (or not): Modulation of the stability-flexibility balance. *Current Directions in Psychological Science*, *28*(1), 3-9.

Dunn, T. L., & Risko, E. F. (2016). Toward a metacognitive account of cognitive offloading. *Cognitive Science*, *40*(5), 1080-1127.

Engestrom, Y. (2000). Activity theory as a framework for analyzing and redesigning work. *Ergonomics*, *43*(7), 960-974.

Gray, W. D., & Boehm-Davis, D. A. (2000). Milliseconds matter: An introduction to microstrategies and to their use in describing and predicting interactive behavior. *Journal of Experimental Psychology: Applied*, *6*(4), 322.

Gray, W. D., Sims, C. R., Fu, W. T., & Schoelles, M. J. (2006). The soft constraints hypothesis: A rational analysis approach to resource allocation for interactive behavior. *Psychological Review*, *113*(3), 461.

Guiard, Y. (1987). Asymmetric division of labor in human skilled bimanual action: The kinematic chain as a model. *Journal of Motor Behavior*, *19*(4), 486–517. https://doi.org/10.1080/00222895.1987.10735426

Hart, S. G. (2006, October). NASA-task load index (NASA-TLX); 20 years later. In *Proceedings of the human factors and ergonomics society annual meeting* (Vol. 50, No. 9, pp. 904-908). Sage CA: Los Angeles, CA: Sage publications.

Hatano, G., Miyake, Y., & Binks, M. G. (1977). Performance of expert abacus operators. *Cognition*, *5*, 57–71.

Hopko, D. R., Mahadevan, R., Bare, R. L., & Hunt, M. K. (2003). The abbreviated math anxiety scale (AMAS) construction, validity, and reliability. *Assessment*, *10*(2), 178-182.

Hutchins, E. (1995). *Cognition in the wild*. MIT press.

Hutchins, E. (2005). Material anchors for conceptual blends. *Journal of Pragmatics*, *37*(10), 1555-1577.

Kirsh, D., & Maglio, P. (1994). On distinguishing epistemic from pragmatic action. *Cognitive Science*, *18*(4), 513-549.

Kool, W., McGuire, J. T., Rosen, Z. B., & Botvinick, M. M. (2010). Decision making and the avoidance of cognitive demand. *Journal of experimental psychology: general*, *139*(4), 665.

Lai, P. C. (2017). The literature review of technology adoption models and theories for the novelty technology. *Journal of Information Systems and Technology Management*, *14*(1), 21-38.

Luchins, A. S. (1942). Mechanization in problem solving. *Psychological Monographs*, *54*(6).

Mackay, W. E. (1990). *Users and customizable software: A co-adaptive phenomenon* (Doctoral dissertation, Massachusetts Institute of Technology).

Mangalam, M., Fragaszy, D. M., Wagman, J. B., Day, B. M., Kelty-Stephen, D. G., Bongers, R. M., ... & Osiurak, F. (2022). On the psychological origins of tool use. *Neuroscience & Biobehavioral Reviews*, *134*, 104521.

Marangunić, N., & Granić, A. (2015). Technology acceptance model: A literature review from 1986 to 2013. *Universal access in the information society*, *14*, 81-95.

Marathe, S., & Sundar, S. S. (2011, May). What drives customization? Control or identity?. In *Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 781-790).

McLaughlin, A. C., & Byrne, V. E. (2020). A fundamental cognitive taxonomy for cognition aids. *Human Factors*, *62*(6), 865-873.

Neth, H., & Payne, S. J. (2001). Addition as interactive problem solving. In *Proceedings of the Annual Meeting of the Cognitive Science Society* (Vol. 23, No. 23).

Norman, D. A. (1988). *The design of everyday things*. Basic Books.

Norman, D. (1991). Cognitive artifacts. In J. M. Carroll (Ed.), *Designing Interaction: Psychology at the Human-Computer Interface*. Cambridge: Cambridge University Press.

Nikolić, D. (2015). Practopoiesis: Or how life fosters a mind. *Journal of Theoretical Biology*, *373*, 40-61.

Osiurak, F., Lesourd, M., Navarro, J., & Reynaud, E. (2020). Technition: When tools come out of the closet. *Perspectives on Psychological Science*, *15*(4), 880-897.

Osiurak, F., & Reynaud, E. (2020). The elephant in the room: What matters cognitively in cumulative technological culture. *Behavioral and brain sciences*, *43*, e156.

Phillips-Wren, G., & Adya, M. (2020). Decision making under stress: The role of information overload, time pressure, complexity, and uncertainty. *Journal of Decision Systems*, *29*(sup1), 213-225.

Risko, E. F., & Gilbert, S. J. (2016). Cognitive offloading. *Trends in cognitive sciences*, *20*(9), 676-688.

Stone, K. D., Bryant, D. C., & Gonzalez, C. L. (2013). Hand use for grasping in a bimanual task: evidence for different roles?. *Experimental Brain Research*, *224*, 455-467.

Walton, M. E., Kennerley, S. W., Bannerman, D. M., Phillips, P. E. M., & Rushworth, M. F. (2006). Weighing up the benefits of work: behavioral and neural analyses of effort-related decision making. *Neural networks*, *19*(8), 1302-1314.

Weis, P. P., & Wiese, E. (2019). Using tools to help us think: Actual but also believed reliability modulates cognitive offloading. *Human factors*, *61*(2), 243-254.

Wiener, N. (2019). *Cybernetics or Control and Communication in the Animal and the Machine*. MIT press.

APPENDICES

Appendix A

Problem Sets for Studies 1 and 2

ProblemID	ProblemSet	Study 1 Prompt	Study 2 Prompt
Init	Practice	17 X ((20 / 60) + 14) = ?	((4 + 1) X 767) / 638 = ?
P1	Practice	17 / ((20 + 60) + 95) = ?	((4 + 1) X 755) / 966 = ?
P2	Practice	77 / ((24 / 60) + 95) = ?	((4 + 3) / 108) X 729 = ?
P3	Practice	47 - ((19 / 89) - 95) = ?	((6 - 1) / 108) X 729 = ?
P4	Practice	47 - ((19 / 89) - 47) = ?	((4 + 3) / 108) X 729 = ?
P5	Practice	47 - ((19 / 26) X 47) = ?	((6 + 2) X 108) / 729 = ?
Init	A	17 X ((20 / 60) + 14) = ?	((4 + 1) X 767) / 638 = ?
A1	A	17 / ((12 X 60) + 60) = ?	((4 - 3) X 767) / 610 = ?
A2	A	17 + ((44 / 60) + 51) = ?	((4 + 2) X 767) / 658 = ?
A3	A	17 / ((37 X 60) + 63) = ?	((4 - 1) X 767) / 660 = ?
A4	A	17 - ((26 / 60) + 30) = ?	((4 + 2) X 767) / 794 = ?
A5	A	17 / ((53 X 60) + 15) = ?	((4 - 1) X 767) / 696 = ?
A6	A	17 X ((51 / 60) + 61) = ?	((4 + 3) X 767) / 508 = ?
A7	A	17 / ((17 X 60) + 45) = ?	((4 - 2) X 767) / 821 = ?
A8	A	17 - ((14 / 60) + 58) = ?	((4 + 1) X 767) / 285 = ?
A9	A	17 / ((96 - 60) + 45) = ?	((4 - 3) X 767) / 545 = ?
A10	A	17 + ((11 / 60) + 42) = ?	((4 + 1) X 767) / 550 = ?
A11	A	17 / ((18 - 60) + 84) = ?	((4 - 3) X 767) / 506 = ?
A12	A	17 + ((16 / 60) + 59) = ?	((4 + 1) X 767) / 896 = ?
A13	A	17 / ((84 + 60) + 65) = ?	((4 - 3) X 767) / 164 = ?
A14	A	17 + ((14 / 60) + 63) = ?	((4 + 2) X 767) / 158 = ?
A15	A	17 / ((11 X 60) + 93) = ?	((4 - 3) X 767) / 721 = ?
A16	A	17 - ((86 / 60) + 37) = ?	((4 + 1) X 767) / 587 = ?
A17	A	17 / ((34 - 60) + 18) = ?	((4 - 2) X 767) / 882 = ?
A18	A	17 - ((45 / 60) + 51) = ?	((4 + 1) X 767) / 608 = ?
A19	A	17 / ((12 X 60) + 82) = ?	((4 - 2) X 767) / 259 = ?
A20	A	17 - ((18 / 60) + 86) = ?	((4 + 3) X 767) / 164 = ?
Init	B	17 X ((20 / 60) + 14) = ?	((4 + 1) X 767) / 638 = ?
B1	B	96 X ((20 / 33) / 14) = ?	((57 - 10) X 767) / 955 = ?
B2	B	34 X ((20 / 83) X 14) = ?	((57 + 21) X 767) / 178 = ?
B3	B	65 X ((20 / 38) - 14) = ?	((57 - 17) X 767) / 99 = ?
B4	B	52 X ((20 / 86) X 14) = ?	((57 + 14) X 767) / 540 = ?
B5	B	31 X ((20 / 73) + 14) = ?	((57 - 20) X 767) / 542 = ?
B6	B	83 X ((20 / 90) - 14) = ?	((57 + 29) X 767) / 189 = ?
B7	B	32 X ((20 / 38) X 14) = ?	((57 - 20) X 767) / 319 = ?
B8	B	18 X ((20 / 64) / 14) = ?	((57 + 22) X 767) / 878 = ?
B9	B	16 X ((20 / 59) - 14) = ?	((57 - 16) X 767) / 120 = ?
B10	B	68 X ((20 / 69) / 14) = ?	((57 + 22) X 767) / 467 = ?
B11	B	80 X ((20 / 90) X 14) = ?	((57 - 10) X 767) / 221 = ?
B12	B	52 X ((20 / 36) + 14) = ?	((57 + 29) X 767) / 665 = ?
B13	B	49 X ((20 / 88) X 14) = ?	((57 - 27) X 767) / 586 = ?
B14	B	36 X ((20 / 30) / 14) = ?	((57 + 24) X 767) / 936 = ?
B15	B	13 X ((20 / 64) - 14) = ?	((57 - 18) X 767) / 725 = ?
B16	B	92 X ((20 / 31) X 14) = ?	((57 + 20) X 767) / 122 = ?
B17	B	66 X ((20 / 74) - 14) = ?	((57 - 19) X 767) / 359 = ?
B18	B	41 X ((20 / 39) / 14) = ?	((57 + 11) X 767) / 905 = ?
B19	B	91 X ((20 / 42) + 14) = ?	((57 - 20) X 767) / 525 = ?
B20	B	26 X ((20 / 34) - 14) = ?	((57 + 27) X 767) / 987 = ?

Appendix B

Questionnaire Items Pertaining to MBT

The following questions are about the machine building task you just completed.

You may have noticed that the problems varied in different ways. What noteworthy changes or patterns did you notice in the problems you were presented?

You may have noticed that the blocks varied in color. What noteworthy behaviors or properties did you notice about blocks of different colors?

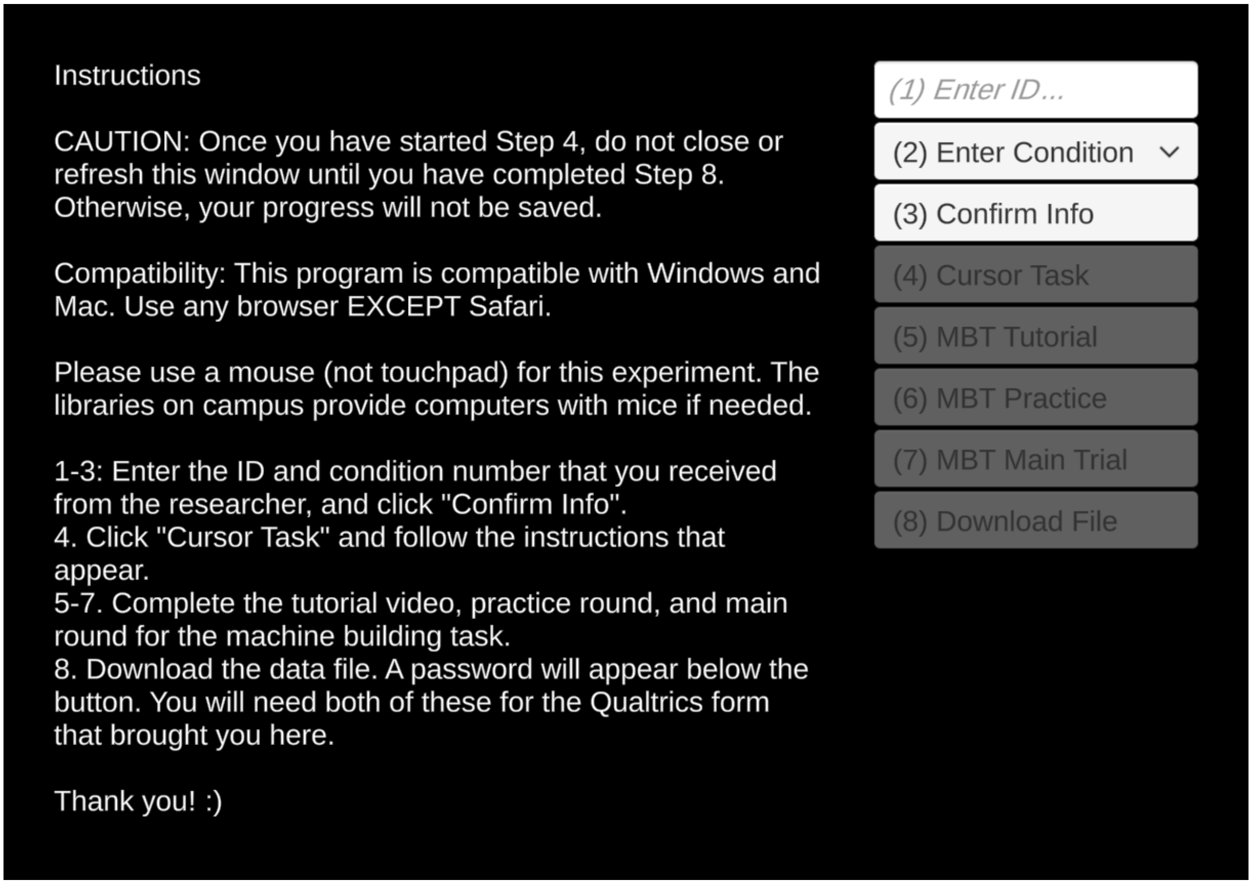
When choosing blocks to build the machines, did you use any particular strategy in the blocks that you chose? If so, describe this strategy.

What other strategies or insights did you find most helpful in completing the problems? Please describe these.

← →

Appendix C

Main Menu of WebGL Program



Appendix D

Disengagements in Study 1. Note: Values indicate count with cumulative duration in parentheses.

Interim			Cursor movement task			Tutorial video			MBT practice trials			MBT main trials		
Idles	Focus losses	Screen exits	Idles	Focus losses	Screen exits	Idles	Focus losses	Screen exits	Idles	Focus losses	Screen exits	Idles	Focus losses	Screen exits
2 (191.9)	1 (3.14)	-	-	-	-	-	-	-	-	-	-	-	-	-
1 (86.37)	1 (9.48)	1 (12.47)	-	-	-	1 (59.51)	-	-	-	-	-	4 (198.77)	-	1 (2.18)
1 (116.74)	-	-	-	-	-	-	-	-	-	-	-	-	-	-
1 (88.8)	4 (21.72)	1 (27.61)	-	-	-	-	-	-	-	-	-	-	-	-
2 (191.56)	-	-	-	-	-	-	-	-	-	-	-	-	-	-
1 (36.17)	-	-	-	-	-	-	-	-	-	-	-	-	-	-
1 (158.51)	-	-	-	-	-	-	-	-	-	-	-	-	3 (0.9)	-
1 (31.83)	1 (1.35)	1 (4.86)	-	-	-	2 (100.82)	-	-	-	-	-	-	-	-
1 (158.92)	-	-	-	-	-	1 (98.02)	-	-	-	-	-	-	-	-
2 (251.54)	5 (43.82)	2 (154.27)	-	-	-	-	-	-	1 (36.52)	-	-	3 (238.03)	-	-
-	-	-	-	-	-	1 (35.95)	-	-	-	-	-	-	-	-
-	-	-	-	-	-	1 (30.14)	-	-	-	-	-	1 (41.18)	-	1 (3.47)
2 (304.18)	-	-	-	-	-	2 (164.38)	-	-	-	-	-	-	-	-
1 (159.14)	-	-	-	-	-	-	-	-	-	-	-	-	-	-
3 (135.99)	-	-	-	-	-	-	-	-	2 (137.55)	-	-	-	-	-
2 (203.91)	-	-	-	-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
2 (564.2)	-	-	-	-	-	-	-	-	-	-	-	-	-	-
1 (87.57)	-	-	-	-	-	1 (38.52)	-	-	4 (174.01)	6 (41.69)	3 (42.81)	-	-	-
1 (147.54)	1 (2.48)	1 (6.9)	-	-	-	-	-	-	-	-	1 (6.18)	-	-	-
1 (156.34)	-	-	-	-	-	-	-	-	5 (269.87)	-	1 (3.47)	-	-	-
1 (158.96)	-	-	-	-	-	-	-	-	-	-	-	-	-	-
4 (394.27)	-	-	-	-	2 (6.44)	2 (108.08)	-	-	-	2 (4.88)	6 (17.22)	10 (451.21)	-	-
1 (74.68)	2 (9.81)	-	-	-	-	1 (81.74)	-	-	-	-	-	-	-	-
2 (199.3)	-	-	-	-	-	-	-	-	-	-	-	-	-	-
-	1 (5.02)	-	-	-	-	1 (37.88)	1 (3.03)	-	-	1 (3.7)	-	-	3 (16.04)	1 (1.53)
1 (159.94)	-	-	-	-	-	-	-	-	2 (92.35)	-	1 (3.72)	-	-	-
3 (493.16)	-	-	-	-	-	-	-	-	-	-	-	-	-	-
4 (416.69)	-	-	-	-	-	-	-	-	4 (414.02)	-	2 (7.86)	-	-	-
-	-	-	-	-	-	4 (282.79)	-	-	-	-	-	2 (70.86)	-	-

Appendix E

Abbreviated Math Anxiety Scale (AMAS; Hopko et al., 2003)

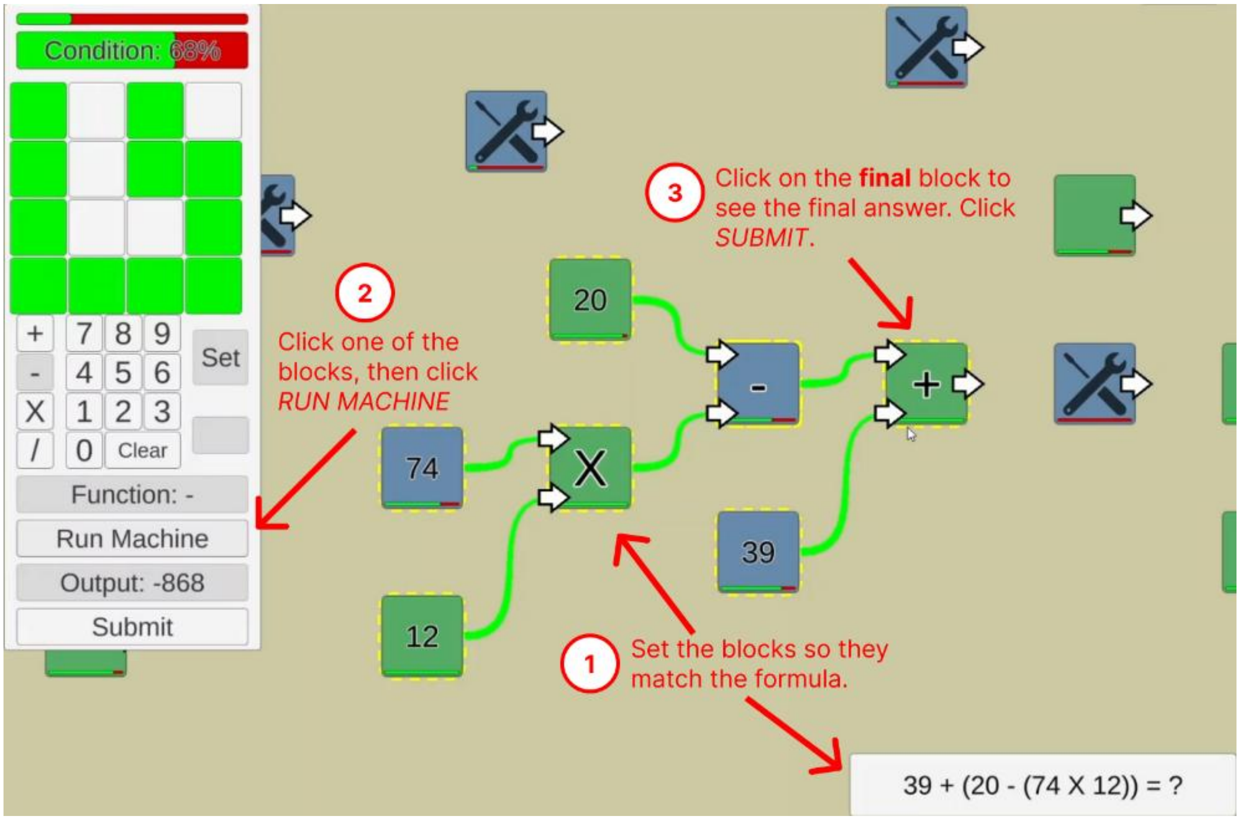
Please rate each item below in terms of how anxious you would feel during the event specified.

	Low anxiety	Some anxiety	Moderate anxiety	Quite a bit of anxiety	High anxiety
Having to use the tables in the back of a mathematics book.	○	○	○	○	○
Thinking about an upcoming mathematics test one day before.	○	○	○	○	○
Watching a teacher work an algebraic equation on the blackboard.	○	○	○	○	○
Taking an examination in a mathematics course.	○	○	○	○	○
Being given a homework assignment of many difficult problems which is due the next class meeting.	○	○	○	○	○
Listening to a lecture in mathematics class.	○	○	○	○	○
Listening to another student explain a mathematics formula.	○	○	○	○	○
Being given a “pop” quiz in a mathematics class.	○	○	○	○	○
Starting a new chapter in a mathematics book.	○	○	○	○	○

← →

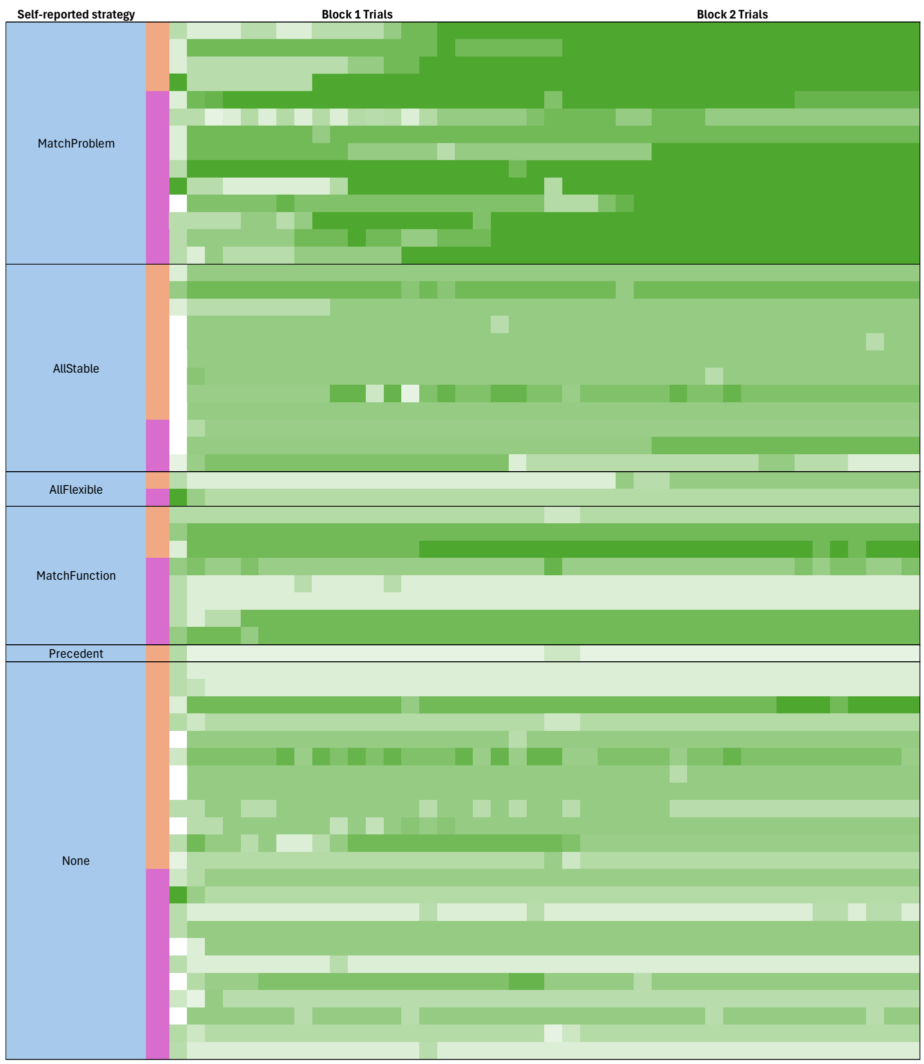
Appendix F

Graphic Aid for MBT in Study 2



Appendix G

Machine Alignment by Trial in Study 2. Note: Alignment between the blocks used in the solving computation and the properties of the problem segments was calculated by summing the flexible blocks used for volatile problem segments and the stable blocks used for static segments, and then dividing the sum by the total number of blocks used. Each row shows the sequence of trials in temporal order from left to right. Greater intensity of green indicates greater alignment. Orange = condition 1, purple = condition 2.



Appendix H

Self-Reported Strategies in Study 2

Level	Description	Example	Count
Action sequences (excluding mental arithmetic)	Coordinating sequence of perceptual, motor, and cognitive actions.	"Instead of waiting for the condition to run out I would just maintain them as close to 100% so it would be less work at one time."	23
Cognitive strategy	Deciding on whether to use mental arithmetic.	"Quick mental math saved adding extra steps to the machine."	24
Resource selection	Selecting blue vs. green blocks.	"I used only green blocks because all the problems would only change 3 variables every new question, so having all the blocks lose durability overtime seemed inefficient."	37
Configuration	Spatial placement of blocks, use of single machine vs. swapping out blocks.	"I visually had my machine laid out in an easy to follow pattern and could reference the same blocks when needing to change elements."	25
Metastrategy	Deciding on what strategy to learn or whether to spend time learning a strategy.	"[…] I learned the behaviors of the different blocks as time went on but I did not want to use excess time to set up a strategy with the information I learned."	1

Appendix I

Disengagements in Study 2. Note: Values indicate count with cumulative duration in parentheses.

Interim			Cursor movement task				Tutorial video				MBT practice trials				MBT main trials		
Idles	Focus losses	Screen exits	Idles	Focus losses	Screen exits		Idles	Focus losses	Screen exits		Idles	Focus losses	Screen exits		Idles	Focus losses	Screen exits
4 (535.0898)	2 (17.60992)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
1 (42.1864)	-	-	-	-	-	-	1 (52.2057)	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
1 (115.6978)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
1 (128.5645)	-	-	-	-	-	-	1 (70.782)	-	-	-	1 (39.83759)	-	-	-	-	-	-
-	3 (26.95)	2 (31.78)	-	-	-	-	2 (194.9549)	-	-	-	-	-	-	-	-	-	-
1 (62.70709)	-	-	-	1 (2.279999)	-	-	1 (39.94231)	-	-	-	-	9 (23.97992)	3 (26.30994)	-	-	-	-
1 (39.3131)	-	-	-	-	-	-	2 (132.9721)	-	-	-	2 (102.7975)	-	-	-	2 (94.25147)	-	-
-	-	-	-	-	-	-	1 (34.85892)	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-	1 (30.18799)	-	1 (4.919922)	-	-	-	-
3 (152.518)	1 (23.39999)	1 (41.12)	-	-	-	-	-	-	-	-	1 (33.35999)	-	-	-	-	-	-
2 (74.43291)	-	-	-	-	-	-	1 (37.35629)	-	-	-	-	-	-	-	-	-	-
-	1 (2.48999)	1 (5.960022)	-	-	-	-	3 (138.1082)	-	-	-	3 (184.7206)	1 (1.289917)	-	-	3 (422.5132)	2 (31.0498)	1 (16.7002)
4 (209.2599)	1 (7.090004)	-	-	1 (0.02999878)	1 (1.860001)	-	-	-	-	1.8	-	-	-	-	-	-	-
-	-	-	-	-	-	-	1 (176.1896)	-	-	-	-	-	-	-	-	1 (0.4599609)	-
1 (91.5527)	-	-	-	1 (0.7400055)	-	-	-	-	-	-	-	-	-	-	-	-	-
5 (689.287)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
1 (151.8361)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
-	4 (80.20001)	2 (98.09999)	-	-	-	-	1 (188.076)	-	-	-	-	-	-	-	-	2 (0.7999268)	-
4 (297.1799)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
1 (104.0096)	2 (15.56001)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
3 (251.7143)	-	-	-	3 (21.48999)	4 (61.35002)	-	-	-	-	61.	10 (720.5085)	-	-	-	1 (76.99268)	-	-
-	-	-	-	-	-	-	2 (85.13031)	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
1 (50.3428)	-	-	-	-	-	-	1 (70.74551)	1 (0)	-	-	-	-	-	-	-	-	-
1 (199.4542)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
1 (198.9892)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
1 (39.56)	-	-	-	-	1 (2.970001)	-	2 (81.7688)	-	-	2.9	-	-	-	-	6 (459.4474)	5 (107.4399)	2 (222.3601)
1 (181.358)	1 (1.139893)	1 (7.589966)	-	-	-	-	-	1 (5.070007)	1 (11.77)	-	-	-	-	5.0	-	-	-
1 (131.331)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
1 (195.5127)	-	-	-	-	-	-	1 (163.9076)	-	-	-	-	-	-	-	-	-	-
1 (91.26559)	-	-	-	-	-	-	2 (75.41452)	-	-	-	-	-	-	-	-	-	-
2 (424.1018)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	1 (4.110001)	-	1 (39.77283)	1 (1.349976)	3 (6.420044)	4.1	-	-	-	1.3	-	-	-
-	-	-	-	-	-	-	1 (33.41931)	-	1 (1.899994)	-	-	-	1 (1.48999)	-	-	-	-
6 (794.7835)	-	-	-	2 (10.18)	1 (18.33)	-	-	-	-	18.	1 (233.3702)	1 (6.670166)	-	-	1 (340.6648)	1 (1.97998)	1 (5.51001)
2 (209.4343)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
3 (298.6504)	4 (63.81004)	-	-	-	-	-	-	1 (2.170013)	-	-	1 (33.02039)	1 (5.779968)	1 (41.53998)	2.1	-	-	-
2 (271.4253)	3 (7.529999)	1 (4.619995)	-	-	-	-	-	1 (10.97998)	2 (21.13998)	-	-	-	-	10.	-	-	-
1 (128.5157)	-	-	-	-	-	-	1 (62.2847)	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	1 (64.2706)	-	-	-	-	-	-	-	-	-	-
2 (247.7765)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
1 (50.633)	-	-	-	-	-	-	1 (144.684)	-	-	-	-	-	-	-	-	-	-
1 (45.5746)	-	-	-	-	-	-	-	-	-	-	-	-	1 (2.279968)	-	-	-	-
2 (64.80799)	-	-	-	-	-	-	1 (44.50549)	-	-	-	-	-	-	-	1 (50.48584)	-	-
1 (84.93127)	1 (4.449951)	-	-	-	-	-	-	-	-	-	-	2 (12.12006)	-	-	7 (436.751)	10 (337.86)	2 (146.8701)
2 (238.5924)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
1 (95.20139)	-	-	-	-	-	-	1 (82.45941)	-	-	-	-	-	-	-	-	-	-
1 (199.4603)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	1 (30.2814)	-	-	-	-	-	-	-	1 (31.25854)	-	-
1 (56.22699)	2 (31.7)	2 (45.42999)	-	-	-	-	2 (99.06216)	-	-	-	-	-	-	-	-	-	-
1 (173.0432)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
1 (198.78)	2 (4.420002)	1 (6.240002)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
-	1 (3.270004)	-	-	-	1 (1.329994)	-	3 (244.2997)	-	-	1.3	-	-	-	-	-	-	-
1 (63.70679)	-	-	-	-	-	-	1 (40.02519)	-	-	-	-	-	-	-	-	1 (3.970093)	-
1 (129.9954)	-	-	-	-	1 (1.549999)	-	1 (68.7305)	1 (0)	-	1.5	-	-	-	-	-	-	-
1 (190.4357)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
1 (200.8344)	-	-	-	-	-	-	-	-	-	-	-	1 (3.440002)	1 (6.440002)	-	-	-	-
1 (201.5717)	-	-	-	-	-	-	-	-	-	-	-	1 (0.8400269)	-	-	-	2 (23)	-
1 (180.9509)	3 (20.64001)	1 (26.26)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-