

# An Open-Source, Fully Customizable 5-Choice Serial Reaction Time Task Toolbox for Automated Behavioral Training of Rodents

Julia Morais Gancz<sup>1</sup>, Nada El Jundi<sup>2</sup>, Eva Strippelmann<sup>2</sup>, Michael Koch<sup>2</sup>, Detlef Wegener<sup>1</sup>

<sup>1</sup>Cognitive Neurophysiology Department, Brain Research Institute, University of Bremen <sup>2</sup>Neuropharmacology Department, Brain Research Institute, University of Bremen

## Corresponding Author

Detlef Wegener

dwegener@uni-bremen.de

## Citation

Morais Gancz, J., El Jundi, N., Strippelmann, E., Koch, M., Wegener, D. An Open-Source, Fully Customizable 5-Choice Serial Reaction Time Task Toolbox for Automated Behavioral Training of Rodents. *J. Vis. Exp.* (179), e63385, doi:10.3791/63385 (2022).

## Date Published

January 19, 2022

## DOI

10.3791/63385

## URL

jove.com/video/63385

## Abstract

The 5-choice serial reaction time task (5-CSRTT) is a behavioral test often used to study visuospatial attention and impulsiveness in rodents. The task requires animals to allocate attention to a horizontal array of five small apertures equipped with light sources and, within a limited time window, nose-poke one illuminated target aperture to get a food reward at the food magazine located in the opposite wall of the chamber. The task considers behavioral control measures such as response accuracy and reaction times and allows to infer selective attention and impulsivity. Task difficulty can be controlled by modifying the stimulus duration and task design in general. Commercially available apparatus usually consists of an experimental chamber and particular software to specify task parameters, but due to fixed hard- and software, they pose many limitations on changes in the general experimental design and specific task requirements and the related data output. This article explains a fully customizable alternative based on an easy-to-use single-board microcontroller and standard electrotechnical components, an open-access Arduino script, and a Matlab-toolbox for hardware control and behavioral task specifications, respectively. The toolbox includes an optional staircase procedure, enabling automated behavioral training. The complete hardware setup, which can be installed in customized chambers, and the freely adaptable software encourage non-standardized task and chamber design. The design of the system and the open-source code for hardware control and experimental setup are described.

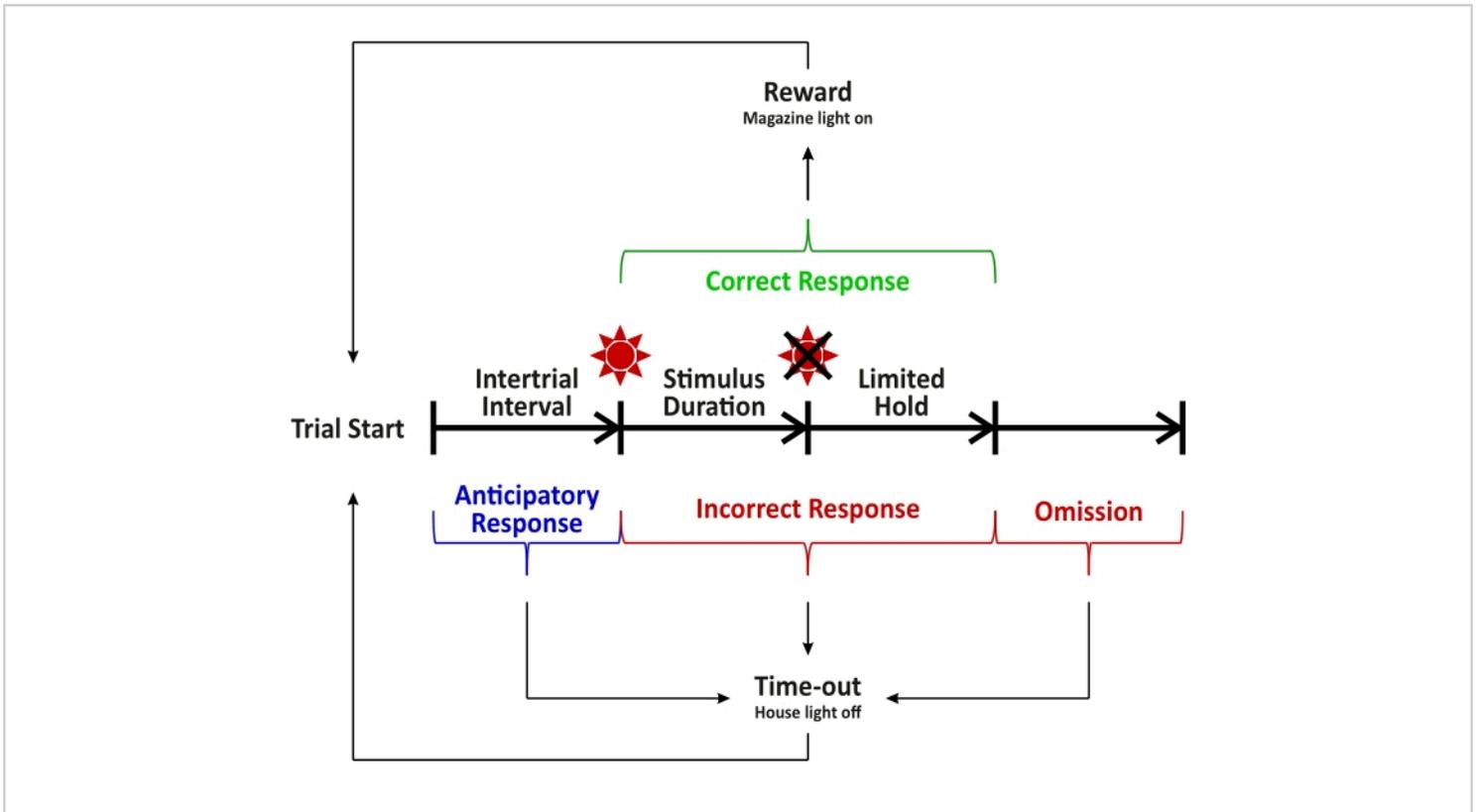
## Introduction

The 5-CSRTT is a behavioral test, often used in rodents to study visual attentional processes and impulsivity<sup>1,2,3,4,5,6</sup>,

such as determining the cholinergic system's role in attention and the influence of norepinephrine reuptake inhibitors

on impulsive behaviors<sup>7</sup>. The standard apparatus allows for observing various control measures such as response accuracy, reaction times, impulsive and compulsive behavior, motor ability, and motivation<sup>1,2,3,4,5</sup>. It consists of a horizontal array of five LED-equipped apertures, a food magazine on the chamber walls opposing the apertures, and house lights<sup>2,5</sup>. In a typical task, the house light is illuminated, and the beginning of a session is marked by the illumination of the food magazine, where a free pellet is delivered. The trial course is initiated when the animal nose-pokes the magazine to retrieve the pellet<sup>1</sup>. After that, the food magazine light is extinguished, and the inter-trial interval (ITI) starts, during which the animal is supposed to direct its attention toward the apertures. Once the ITI elapses, a stimulus illumination is presented in one of the apertures<sup>1,2,5</sup>. The stimulus is

given for a specific length known as the stimulus duration (StD). The animal can respond to the stimulus while it's being presented or during a limited time window after the StD is over, known as limited hold (LH). To respond, the animal has to nose-poke the target aperture and, if done correctly, a reward is released in the food magazine<sup>1,2,5</sup>. Otherwise, any incorrect response, as well as any responses before stimulus presentation (anticipatory or premature) or any failure to respond (omission) results in a timeout (TO), during which the house light is turned off for a certain duration<sup>1,2,5</sup> (**Figure 1**). In general, the discriminative stimulus accuracy measures attentional functioning, whereas premature and perseverative responses (repeated responses at the apertures after stimulus presentation) are considered measures of impulsive behavior and compulsivity, respectively<sup>1,4,5,6</sup>.



**Figure 1: Possible trial sequences of a typical 5-CSRTT.** Following the Intertrial Interval, the stimulus light is turned on for a specific duration, and then turned off during the Limited Hold interval. The rat can either answer correctly and receive a reward or answer incorrectly and get a timeout during this time. If the rat does not answer in time, its omission results in a timeout. Similarly, if it responds before the presentation of the light stimulus, its premature response results in a timeout. Another trial starts after the collection of the reward or the end of the timeout period. [Please click here to view a larger version of this figure.](#)

The 5-CSRTT is widely used due to its flexibility: by changing parameters of the trial design, different subcategories of attention can be investigated. For example, while the animal is supposed to divide its attention across the five different apertures (visuospatial attention), the use of irrelevant stimuli (e.g., auditory stimuli) enables the testing of selective or sustained attention<sup>1,2,5,6</sup>. For that purpose, the experimental setup can be extended by including speakers, which can be used as distracting or even reinforcing stimuli<sup>1,2,5,6</sup>. Moreover, the attentional load can be directly modulated by altering the stimulus presentation or randomizing the ITI duration<sup>10</sup>. The 5-CSRTT is not only used in rodents<sup>3,7</sup>, but has recently been adapted to test non-human primates<sup>1,7,8</sup> and fish<sup>7,9</sup>, further showcasing its feasibility. A fully customizable 5-CSRTT toolbox allows for easy adaptation of the standard rodent paradigm to other animal models. Moreover, the customization flexibility of the 5-CSRTT toolbox also encourages research using non-standard task designs.

The fully customizable 5-CSRTT toolbox presented here contains an Arduino script for hardware control, programmed in an integrated development environment. It also consists of a Matlab toolbox (version R2019b or younger) for experiment control. The following protocol explains how to configure the 5-CSRTT toolbox with the widely used standard paradigm and shows optional configurations for non-standard paradigms.

## Protocol

The experimental procedure in this protocol was performed following the recommendations of the EU directive 2010/63 for the Welfare of Experimental Animals and in accordance with the Animal Welfare Act issued by the Federal Government of Germany and was approved by the local authorities. Since the research only required behavioral training, no animals were euthanized, and all were kept in the husbandry after the research was conducted. The research was conducted using ten male Lister hooded rats (4 months of age at the start of behavioral training).

### 1. Animal housing, husbandry, and handling

1. House rats together with up to five littermates in a standard cage with suitable bedding material, according to your animal welfare committee's recommendations.
2. Keep the rats in a ventilated room with a controlled temperature of  $20 \pm 2$  °C and relative humidity of a maximum of 50% with a 12:12 h light/dark cycle, or in accordance with your animal welfare committee's recommendations. Restrict access to food (12 g chow a day per rat, see **Table of Materials**) and provide unrestricted access to water.
3. Mark the animals' tails using a permanent non-toxic ink marker.

4. Before starting the behavioral experiment, handle the rats for at least a week until they are accustomed to being handled by experimenters, and introduce the rats to the rewarding food pellets to reduce food neophobia.

## 2. Preparation of the hardware and control software

1. Open the freely available Integrated Development Environment (IDE) Software (see **Table of Materials**). Click on **File > Open** and click on the script for hardware control (**Supplementary File 1**).
2. Plug in the microcontroller USB to the computer. Check whether the automatically chosen Board and Processor information correspond to the connected microcontroller board by clicking on **Tools** in the top left half of the screen. Select the corresponding Board and Processor information, and click on **Port** to select the available port.
3. On the top left half of the screen, click on **Sketch > Include Library > Manage Libraries**. On the newly opened Library Manager window, type the term "Adafruit Motor Shield V2" into the search bar and click on the **Install** button on the corresponding library. Repeat the same process for the search term "Adafruit Neopixel".
4. On the top left half of the screen, click on **Verify** (the button with a checkmark) to ensure there are no mistakes in the script. Click on **Upload** (the button with an arrow to the right) to upload the script to the microcontroller board.

## 3. Preparation of the experiment control software

1. Ensure that all four scripts and functions for experiment control are located in the same folder.
  1. Open the programming platform, click on the **HOME** toolbar tab on the top half of the screen, and click on

**Set Path.** Click on **Add Folder** and select the folder containing all the experiment control scripts.

2. Click on **Save** and close the **Set Path** window. Click on **Open** on the **HOME** toolbar tab on the top half of the screen and open the following scripts and functions: **User (Supplementary File 2)**, **Staircase (Supplementary File 3)**, and **DataProc (Supplementary File 4)**.
2. Follow the instructions for downloading and installing the Psychtoolbox software to enable the ESC button functionality used by the toolbox (see **Table of Materials** for the access link).

## 4. Configuration of parameters used in the 5-CSRTT toolbox

1. Prepare the toolbox for habituation.
 

**NOTE:** **Figure 2** represents the 5-CSRTT apparatus used for the current study.

  1. Select the opened **User** script. Ensure that the habituation variable on line 7 is set to 'true'. Write a number in minutes (e.g. '30' for 30 min) on line 8 to set a time limit for the habituation. On line 9, write a number between 0.01 and 1 (full brightness) to choose a brightness level for the stimulus light.
 

**NOTE:** The brightness level used in this project is set to 0.2.
  2. Conduct the behavioral experiment (step 5).
2. Prepare the toolbox for the experiment session.
  1. Select the opened **User** script. Ensure that the variable on line 7 is defined as 'false'.
  2. On line 12, type in the directory path where the experiment data will be automatically saved (e.g.,

'C:\Users\trainer\Desktop\5CSRTT'). Ensure that a directory with this exact name exists.

3. Type in the subject identification on line 13 (e.g. 'red1').
4. Ensure that the variable on line 14 is defined as 'true', so that the data generated is automatically saved in the directory path.
5. Ensure that the variable on line 15 is defined as 'true' and type in a number on lines 16 and 17 to set a trial and time limit (in min), respectively, after which the program will automatically stop.
6. Configure the parameters for a standard 5-CSRTT paradigm.
  1. Check whether the variable on line 32 is set to 'true' for a free pellet delivery before the first trial. Define the variable on line 33 as 'true' to ensure a magazine entry before each trial starts. Check whether the variable on line 34 is set as 'defined' and type in a number on line 35 to specify the length of the ITI in seconds.
  2. Ensure that the variable defined on line 37 is set to 'true' so that trials with premature answers won't influence the trial limit of the session. Type a numerical value on line 38 that will define the limited hold (LH) length in seconds.
  3. Ensure that the variable on line 39 is set to 'none' and the variable on line 40 is defined as 'false'. Type a number on line 41 to define the TO length in seconds and ensure that the variable on line 42 is set to 'false'.
  4. Type in a numerical value on line 45 that corresponds to the number of apertures that can

be illuminated (e.g., '5'). Ensure that the variable on line 46 is defined as 'pseudorandom' or 'random' and type in a number on line 47, setting the StD length in seconds. Check whether the variable on line 48 is defined as 'single' and the numerical value on line 49 is '1'. On line 50, type in a number to define the brightness of the target stimulus.

5. Check whether the variables on lines 64 and 65 are defined as 'binary' and 'non-dependent', respectively. Type a numerical value on line 71 corresponding to the number of food pellets to be released following a correct nose-poke.
6. Conduct the behavioral experiment (step 5).
7. Configure the parameters for a non-standard 5-CSRTT paradigm.
 

**NOTE:** All the steps described in this subchapter are optional.

  1. If an automatic performance check is desired for using the automated staircase training procedure, ensure that the variable on line 18 is defined as 'true' and type in a numerical value on line 20 defining the frequency of the performance check. Type a number on line 21 to define the minimum number of trials to be completed during the current session before calculating the subject's performance. Ensure that the variable on line 21 is defined as 'all' so all trials of the current session are included in the performance check.
  2. Ensure that the variable on line 22 is set to 'true' so that the program will update the parameters of the current session to match a previously

completed session. On line 23, type in 'latest' to establish that the last session will be uploaded.

**NOTE:** The program will update the parameters based on specifications in the "Staircase" function, jumping to the previously completed level. A specific dataset to be uploaded can also be chosen by typing in the exact path to the data file with ".mat" ending.

3. If the automated behavioral training is desired, ensure that the variable on line 26 is set to 'true'. Type in a numerical value on lines 27 and 28 to define the training level to start with and the total number of levels available, respectively. If a division of cohorts is desirable, type in a name (e.g., 'group1') on line 29 that specifies the group.

**NOTE:** Each group can use its own set of training levels and criteria for level updates. The parameters for each group are defined in the "Staircase" function (step 4.2.8.).

4. On line 34, type in 'random' if a randomized ITI duration is desired. Type in a numerical interval (e.g., '[0,2]') to define the interval containing a randomized number that will be added to the fixed ITI duration.
5. To ensure that trials with premature responses influence the session's trial limit, type in 'false' on line 37.
6. To define a time window during which additional nose-pokes will be counted as perseverative answers, type in a numerical value on line 39. Type in 'true' on line 40 so that premature answers evoke a timeout.

7. To define different groupings of the target apertures, type in 'neighbour', 'shifted' or 'all' on line 48. Type in a numerical value on line 49, defining the total number of target apertures. If dimmed stimuli are desired, type in a numerical value on lines 51 and 52, defining the total number of dimmed apertures and its brightness, respectively.
8. If the emission of a short tone (Tone C4, 262 Hz (Scientific Pitch Notation)) before stimulus presentation is desired, ensure that the variable on line 55 is set to 'true'. Type in numerical values on lines 56, 57, and 58 to define the time window (in milliseconds) between the speaker's tone and stimulus presentation, the tone's duration (in milliseconds), and the tone's volume (numbers between 0 (no tone) and 1 (full volume) are allowed).
9. If the emission of a short tone (Tone C6, 1047 Hz) after a successful nose-poke is desired, ensure that the variable on line 59 is set to 'true'. Type in a numerical value on lines 60 and 61 that define the tone's duration (in milliseconds) and the tone's volume (numbers between 0 (no tone) and 1 (full volume) are allowed).
10. If responses in dimmed lit apertures are to be rewarded, ensure that the variable on line 64 is set to 'non-binary'. Type a numerical value on line 73 to define the number of food pellets delivered for nose-pokes in dimmed lit apertures.
11. If using more than one pellet dispenser, type in the corresponding motor number on lines 70

and 72 for nose-pokes in target apertures and dimmed lit apertures.

**NOTE:** The motor number can be either 1 or 2. The corresponding motor shield's screw terminals M3 and M4 are defined in the script for hardware control.

12. If grading the reward based on reaction time is desired, ensure that the variable on line 65 is set to 'dependent'.
  1. Define the dependence of nose-pokes in target apertures by typing in numerical values on line 67 that will divide the reaction time (in seconds), motor number, and the number of food pellets to be delivered into different categories, such that a specific reaction time interval corresponds to a chosen motor number and number of pellets.
  2. Type in numbers on line 68 to define different categories for reaction time (in seconds), motor number, and number of food pellets to be delivered for nose-pokes in dimmed lit apertures.
  
8. Configure the Staircase function by following the steps below.
 

**NOTE:** This step is optional.

  1. Select the opened **Staircase** function. On line 4, type the name of the first group (e.g. 'group1'). If applicable, type the name of the second group (e.g. 'group2') on line 77.
  2. To change the parameters for the second training level for the first group, type one of the parameters calculated in the

performance check on line 17 (e.g., PerformanceCheck.NumCorrect >= 30 if the criteria are answering 30 nose-pokes correctly).

**NOTE:** Do not change the parameter "Config\_trigger == 2" when using the automatic loading of the previous session (step 4.2.7.2.).

3. On line 19, type a variable that you wish to update and a numerical value if applicable (e.g., 'Config.LED.StimDuration = 30' to set the StD length to 30 s).
 

**NOTE:** The number of parameters to be changed and their new value can be freely chosen. The only requirement is that the parameter to be updated needs to be typed in after the variable 'UpdateTrigger = 1' in each level that the update is desired.
  
9. Configure the "DataProc" function.
  1. Select the opened **DataProc** function. If a graph with the session's overview needs to be plotted and saved automatically, type in the commands for the desired plot from line 83 onward.
 

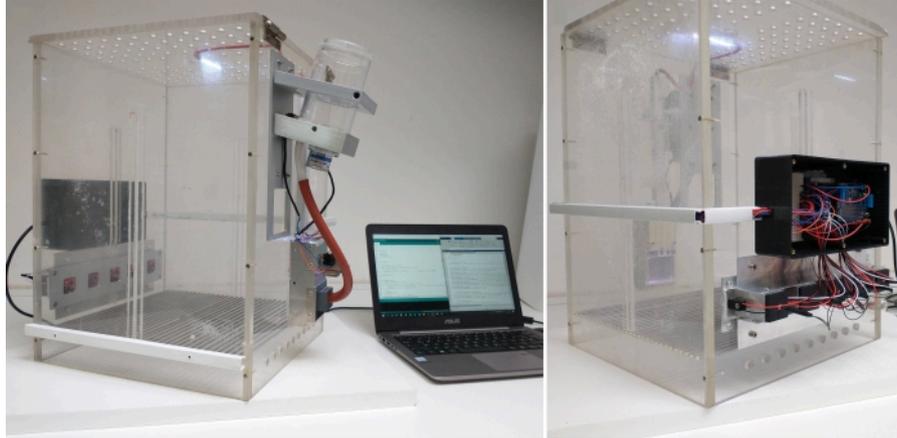
**NOTE:** The current commands on line 83 onward plot an overview of the session's outcome and some necessary control measures such as the total number of premature answers or the number of food panel pushes during an ITI.

## 5. Behavioral experiment

1. Transport the rat cage from the vivarium to the experimental room at least 30 min before the habituation or experiment session to familiarise the animals to the testing room.

2. For the habituation session, prepare the operant chamber by placing two reward food pellets in each of the apertures and five food pellets in the magazine door. Configure the parameters for the session by following steps 2-4.1.  
**NOTE:** The operant chamber used for this protocol was a modified Skinner PVC box with dimensions 30 x 30 x 45 cm.  
**NOTE:** For the first habituation stage, tape the magazine flap door to remain open. For the second habituation stage, remove the tape on the magazine flap door.
3. Select the opened **User** script. Ensure that the 'COM' description on line 75 matches the chosen available port in step 2.2. If it doesn't match, change the numerical value in the experiment control script (e.g., from 'COM3' to 'COM4').
4. Gently place the rats in the chamber.
5. In the opened User script, click on the **EDITOR** toolbar tab on the top half of the screen, and then click on the green play button **Run**. Check whether the program is running correctly by reading the "Command Window" information.
6. To stop the experiment at any time, press the escape key **ESC** on the computer's keyboard. Wait for a message to appear on the Command Window display. Type in "y" and press the **Enter** key on the computer's keyboard to stop the current session and save the acquired data.
7. When the habituation or session time or trial limit is reached (step 4.1.1 or 4.2.5, respectively), check the message that appears on the Command Window display. Type in "y" and press the **Enter** key on the computer's keyboard to stop the current session.  
**NOTE:** The message will only be shown at the beginning of a new trial and will stop the ongoing session until an answer is typed in ("y" to stop the session or "n" to continue the experiment).
8. In the case of a habituation session, check whether the rat consumed all the food pellets. Repeat the habituation stage until all the pellets are consumed before advancing to the next habituation stage or, after the second stage, start the 5-CSRTT training.
9. After the session, clean the walls and floor of the operant chamber, e.g., with a 70% ethanol solution and paper towel. Before introducing the next rat, wait for 2-3 min until the ethanol smell dissipates.
10. When the experiment day is over, unplug the microcontroller USB from the computer. Optionally, close both scripts for hardware and experiment control.

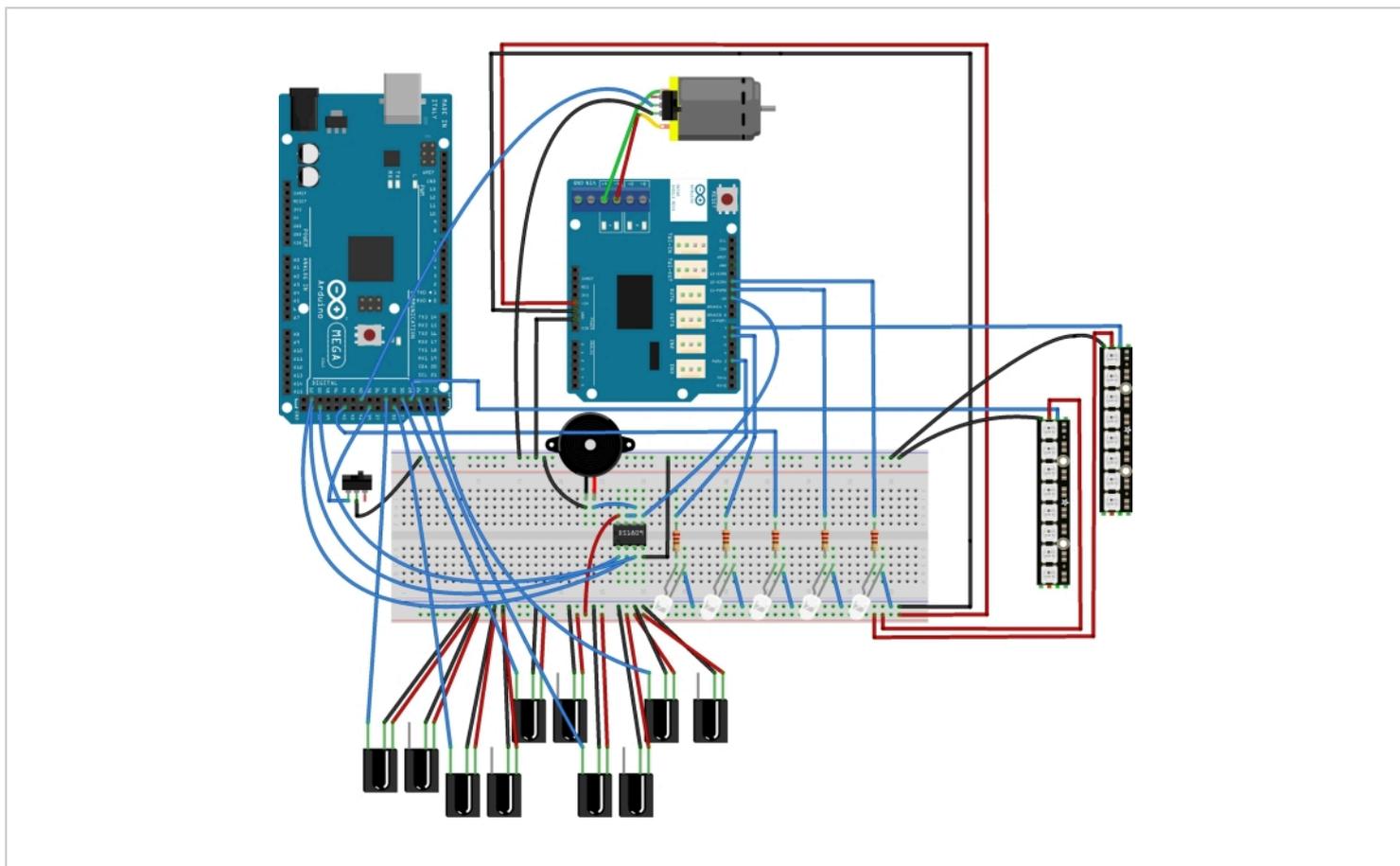
## Representative Results



**Figure 2: The 5-CSRTT apparatus used for the current study.** The apparatus runs on a laptop equipped with the 5-CSRTT toolbox, which provides a script for controlling the microcontroller and all related equipment and multiple scripts for controlling the 5-CSRTT experiment. [Please click here to view a larger version of this figure.](#)

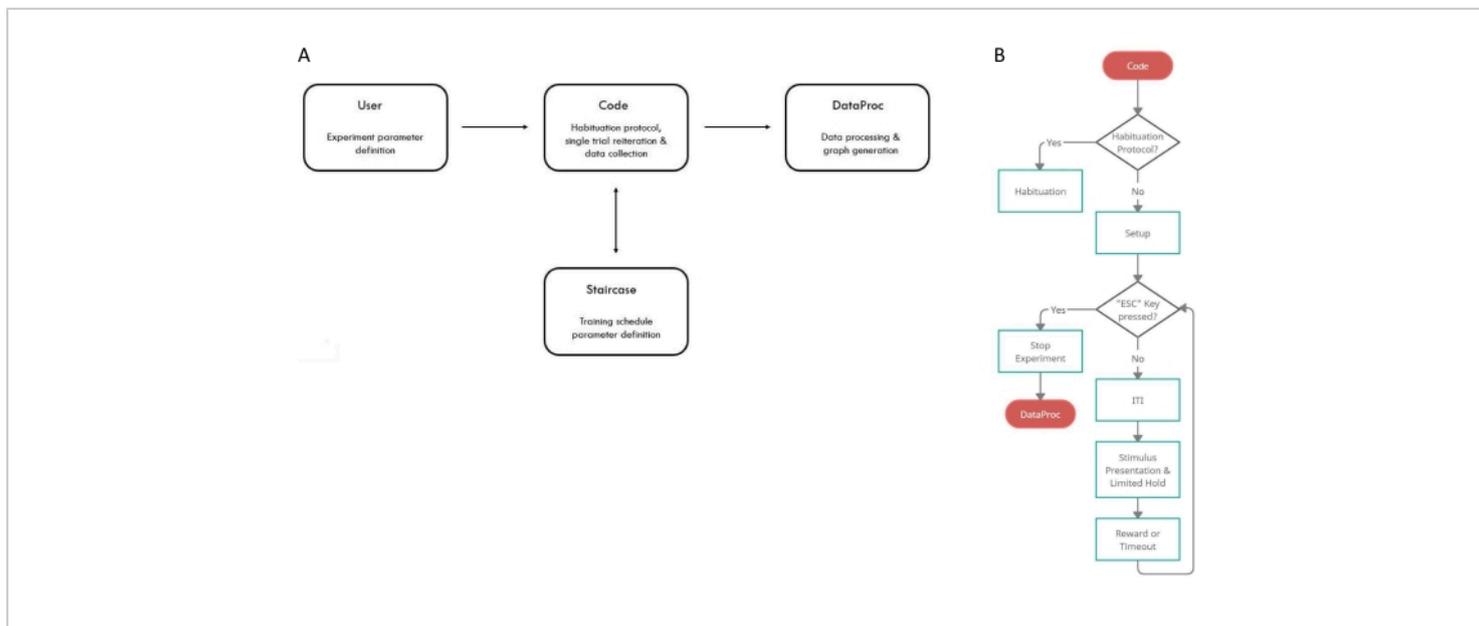
The fully customizable toolbox is easy to use and based on a single-board microcontroller and standard electrotechnical components. **Figure 3** shows a simplified circuit and wiring diagram. The whole aperture consists of 5 LEDs as the light stimuli and five infrared sensors to detect nose pokes. The house light consists of one strip with eight LEDs, and the food magazine is made out of an aperture with a flap

door with a micro switch, a motor-driven pellet dispenser, and a strip with eight LEDs for lighting. The circuit also exemplifies connections for optional components such as the passive buzzer speaker for auditory feedback and a digital potentiometer for volume adjustment. For a list of the equipment used in developing this toolbox, please see **Table of Materials**.



**Figure 3: Simplified circuit of the microcontroller hardware.** To be easily and quickly customizable, the microcontroller equipment is connected *via* a breadboard. From top left to bottom left, clockwise: A microcontroller board is connected to a motor shield and a DC motor (representing the pellet dispenser motor). To the right are the LED strips for both the house and food magazine lights, and in the middle are all five white LEDs for the stimulus light and the five infrared sensor pairs used in the apertures. Below the microcontroller board is a simple microswitch (representing the switch used in the food magazine

flap door). Finally, a passive buzzer speaker and a digital potentiometer are depicted in the middle. This image was made using the open-source software Fritzing. [Please click here to view a larger version of this figure.](#)



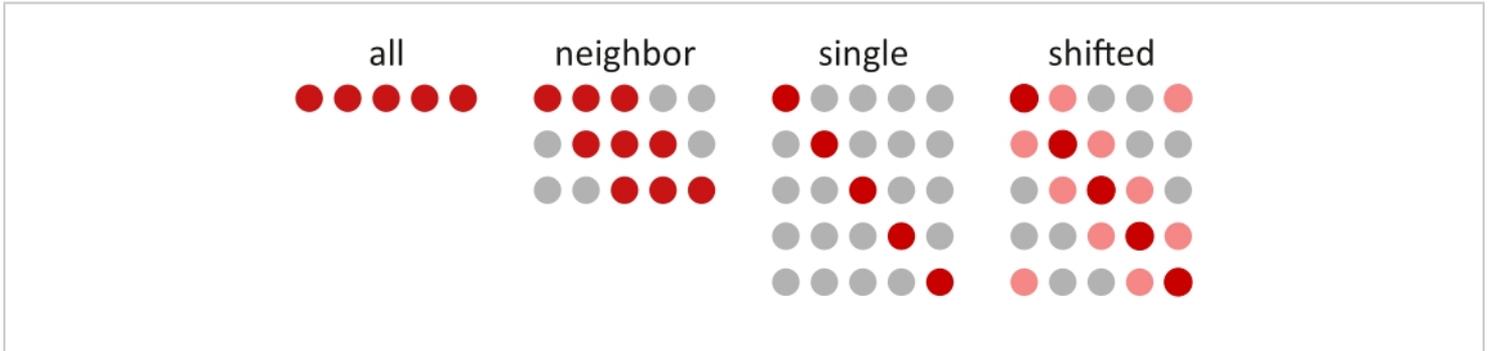
**Figure 4: Linkage and functions of all components of the experiment control scripts and simplified diagram of the "Code" function. (A)** The "User" script sends its parameters to the "Code" function, which in turn links directly to the "Staircase" function, allowing it to update any parameter used in the "Code" function while the experiment is ongoing. The "Code" function then sends its results to the "DataProc" function at the end of the session. **(B)** Before starting an experiment session, the "Code" function first checks whether it is supposed to start the habituation protocol. If not, it sets up the parameters based on the definitions chosen in the "User" script. Before each trial begins, the function then checks whether the ESC key on the keyboard was pressed. If not, it continues with a new trial. Otherwise, it stops the experiment session and passes the gathered data to the DataProc function. This critical check before each trial start allows the program to stop before any chosen time limit is reached. [Please click here to view a larger version of this figure.](#)

Interactions between the different experiment control scripts can be seen in **Figure 4A**. The "User" script includes all the parameters that define the experiment. There, variables that determine the experiment's timing, number and brightness of illuminated stimuli, ITI duration, and the like can be freely chosen. The Code function (**Supplementary File 5**) includes a detailed description of a single trial and all possible outcomes, which is reiterated throughout the experiment, as shown in **Figure 4B**. Moreover, it consists of a protocol for

the habituation of the animal to the apparatus. The Code function also regularly checks the performance of the animal. Furthermore, the Staircase function is optional. The subject's performance is compared to previously set criteria, and desired parameters are automatically updated if the animal's performance meets these criteria. The Staircase function may also consider the acquired results from the previous day's session. While the experiment is running, a performance check at the end of a trial will calculate accuracy, omissions,

and the total number of correct responses of the completed trials and compare the outcome with desired criteria for a level update, as specified in the Staircase function. Finally, the DataProc function processes all the gathered data and

generates simple graphs for quick analysis. At the end of a session, the toolbox automatically saves all the data into a \*.mat file and generates an extra \*.xlsx file with the essential information from the experiment.



**Figure 5: Example of different stimulus configurations of the 5-CSRTT toolbox.** The diagram exemplifies possible combinations of target stimuli in dependence on the chosen configuration. Both the "all" and "single" configurations are used in the standard paradigm (for the habituation and behavioral experiment). The "neighbour" and "shifted" configurations show non-standard stimulus configurations, allowing the use of other numbers of lit stimuli, which may also have a different contrast than the target stimulus. [Please click here to view a larger version of this figure.](#)

Protocol step 4.2.7.7 mentions an optional feature: changing the grouping of target apertures. The standard 5-CSRTT paradigm makes use of one single target stimulus. Here, we exemplify how the toolbox presented allows for modifications of the standard paradigm. **Figure 5** displays some possible group combinations out of a total of five apertures concerning the chosen configuration. The "all" configuration lights up all the available apertures so that each aperture is now a target aperture, which can be helpful in the initial training stages. The neighbour configuration makes sure that the (freely chosen) number of target apertures will be neighboring each other. Settings can be specified such that the neighbors will not be identical with the target aperture but be illuminated at lower (or even higher) contrast. The use of apertures with different illumination contrasts allows for testing new paradigms, such as using differently graded rewards for

nose-pokes in the high- or low-contrast apertures. **Figure 5** shows an example with three target apertures with identical illumination. The single configuration is typically used in the standard 5-CSRTT, where only a single target is illuminated. Finally, the shifted configuration extends the neighbor configuration. It shifts the neighbor stimulus toward the last or first aperture in case the target aperture is at the first or last position, respectively. As in the neighbor configuration, the illumination strength of the neighbors can be freely chosen, being either the same or different from the target aperture. Also, the number of overall lit stimuli can be freely chosen. The toolbox then computes all possible stimuli automatically. However, the parameter "Config.LED.NumHighLED" must be set to "1" for this configuration.

Following the protocol, the training of rats (N = 10) for the 5-CSRTT was performed according to the training stages presented in **Table 1**.

**Table 1: 5-CSRTT training schedule and criteria to move to the next level.** (A) The inter-trial interval was kept constant at 5 s in every training level. (B) Stimulus duration for every training level. (C) Limited Hold (LH) time window, the maximum time tolerated between stimulus off and any nose-poke response. (D) The total number of correct responses needed to pass the respective training level. (E) The accuracy percentage is calculated as  $\frac{\#correct}{\#correct + \#incorrect}$ . (F) Percentage of omission errors is defined as  $\frac{\#omission}{\#correct + \#incorrect + \#omission}$ . This criterion does not include premature responses. [Please click here to download this Table.](#)

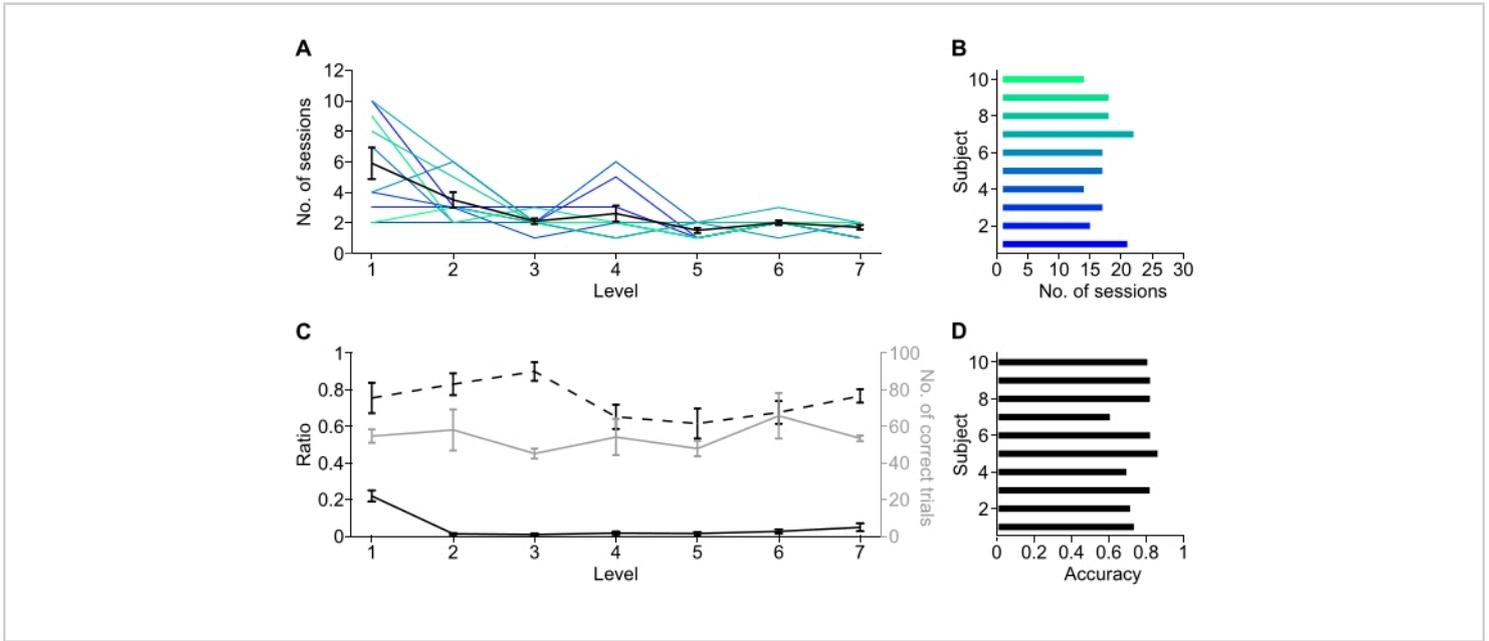
The performance of the rats was compared to the number of training days (sessions) required to complete each training level given in **Table 1**. All the animals started at training level 1 with an StD and LH of 60 s each. However, some rats (N = 5) received enhanced habituation training to test some of the additional stimulus options reported earlier, which explains the difference in the number of sessions the individual animals stayed in training level 1. Completion of the level was marked by reaching a total of 30 or more correct responses. StD and LH decreased during the following levels, while the criteria for advancing to the next training level got tougher, increasing the attentional demand of the task<sup>1,6</sup>.

**Table 2** shows the automatically generated \*.xlsx spreadsheet of one example rat during one session. The rat started with the configuration specified in training level 5. After four trials, the rat advanced to level 6, considering the trials performed in the current session plus the accuracy

achieved in the previous session. How many trials have to be performed at a minimum in the current session to advance to the next training level is specified in the variable "Config.Experiment.MinNumTrials". In the same session, the rat advanced to training level 7 after completing 66 trials in level 6 and achieving the requirement of > 80% accuracy and < 20% omission. In total, rats were trained for 26 days using the configuration of training levels as provided in **Table 1**. The number of sessions spent per training level is provided in **Figure 6A**. The black line shows the average across all subjects, and each colored line displays the data of one rat. All rats reached the eighth level within 14-22 sessions (**Figure 6B**). **Figure 6C** shows the mean performance of subjects per training level and across all training days in the 5-CSRTT apparatus. The dashed black line represents the accuracy percentage, and the straight black line represents the omission percentage. Accuracy was calculated as the ratio between the number of correct responses and the total number of responses. Omissions were calculated as the ratio between the number of omissions and the total number of trials (i.e., the sum of correct responses, incorrect responses, and omissions). The gray line indicates the average total number of correct responses across all trials in each level. **Figure 6D** displays the final accuracy achieved by each subject on the eighth and final training level.

On average, rats spent 5.9 (±1.03 SEM) sessions to complete level 1, between 1.5 (±0.17) and 3.5 (±0.5) sessions to complete level 2 to 6, and 1.7 (±0.16) sessions to complete level 7 before they reached the final level 8. As is evident from **Figure 6A**, the variance between subjects was most significant in the initial levels (SD = 3.25 in level 1, 1.58 in level 2) and decreased in later levels (0.47 and 0.48 in levels 6 and 7, respectively). In level 4, when the stimulus duration was further reduced, the average number of sessions spent

( $2.6 \pm 0.52$ ), and variance between rats (1.64) increased, with two rats taking 5 and 6 days to conclude the level.



**Figure 6: Results of the behavioral experiment with the 5-CSRTT toolbox.** (A) The number of sessions performed at each training level. The black line depicts the average number of sessions of all subjects for each level (mean  $\pm$  SEM), and colored lines represent individual subjects' data. (B) The absolute number of sessions needed to reach the final level, per subject. (C) Averaged performance measures throughout training (mean  $\pm$  SEM). The dotted black line depicts the accuracy of all subjects across all given responses in all sessions per training level, and the black line shows the corresponding omission percentage. The gray line depicts the average absolute number of correct answers of all subjects at each training level. (D) Accuracy per subject during the eighth and final training level. [Please click here to view a larger version of this figure.](#)

**Table 2: Gathered data from one example rat during one training session.** Column A displays the trial count over the session regarding the current training level, as shown in column B. Column C displays the ITI duration, and column D displays the trial start time. Columns E to I show the brightness level for the LED stimulus in apertures 1 to 5, respectively. A brightness level of 0 means the stimulus was off, and a brightness level of 0.2 means the stimulus was turned on with 20% of its maximal intensity. Columns J and

K show the exact time the stimulus was turned on and off, respectively. Column L displays the outcome of the trial: 0 means "omission", 1 means "correct response", 3 means "incorrect response" (nose poke into non-target aperture) and 4 means "premature". Column M shows which aperture was nose-poked during the trial, while column N depicts the exact time of the nose-poke. Columns O, P, and Q show the time when the pellet dispenser motor was turned on, the corresponding motor number, and the time when the rat

opened the pellet dispenser to get its reward, respectively. Column R displays the trial end time. Columns S, T, U, V, and W show the total number of premature responses, timeouts, panel pushes during an ITI, the total number of perseverative answers, and the total runtime of the session in minutes, respectively. [Please click here to download this Table.](#)

**Supplementary File 1: Script for the hardware control of the IDE software (Arduino code).** This includes all commands to control the hardware and electrotechnical components of the toolbox. [Please click here to download this File.](#)

**Supplementary File 2: Script for the function "User" in the experiment control software.** This includes all the parameters that define the experiment. [Please click here to download this File.](#)

**Supplementary File 3: Script for the function "Staircase" in the experiment control software.** This monitors the subject's performance and compares it to the previously set criteria. The desired parameters are automatically updated if the animal's performance meets these criteria. [Please click here to download this File.](#)

**Supplementary File 4: Script for the function "DataProc" in the experiment control software.** This processes all the gathered data and generates simple graphs for quick analysis. [Please click here to download this File.](#)

**Supplementary File 5: Script for the "Code" function.** This includes a detailed description of a single trial and all the possible outcomes, which is reiterated throughout the experiment. [Please click here to download this File.](#)

## Discussion

The present protocol aims to develop and test a low-cost and fully customizable alternative to the standard, commercially available 5-choice serial reaction time task apparatus. Usually, commercially available types of apparatus provide a limited set of features as needed to run the standard research 5-CSRTT. Because of this, non-standard modifications in the specific trial design, such as changes to the trial sequence or target stimulus combinations, are usually not possible. In addition, many of the available types of apparatus come with specific, closed software that may not provide access to all behavioral data of the experiment, such as the timing and aperture number of premature and perseverative answers. In contrast, the vital advantage of the toolbox presented here is - besides its low cost - the possibility of implementing many different trial designs and research paradigms. Currently, the toolbox supports the definition of multiple stimulus designs, such as permitting dimmed stimuli and using two pellet delivery systems and reward dependence on reaction time. It also supports the use of a miniature speaker for auditory feedback. However, the primary purpose is to allow for easy modifications of the trial sequence according to the user's objective, such as introducing dimmed lit apertures and rewarding schedules for decision-making tasks or integrating state-of-the-art positive reinforcement training approaches<sup>11</sup>. Additionally, all raw data acquired over the session are made available for further analysis. The toolbox provides a Staircase function for automated behavioral training, which is also fully customizable and allows the user to change the criteria for each level update, the number of training levels, and the parameters to be updated. Furthermore, the apparatus itself is highly adaptable, and changes in the trial design and the chamber layout are easily feasible, making it possible to apply the 5-CSRTT paradigm to animal

species that need a different research design than what the commercially available types of apparatus offer.

Specific parts of the protocol for the software configuration are critical to ensure a smooth workflow: especially for the first day of the experiment, the preparation of both the hardware and experiment control software (steps 2 and 3) and the configuration of step 5.3 is crucial. Ensuring that the serial port connection between the microcontroller hardware, its software, and the experiment control software is working properly is critical to establishing a fully functioning 5-CSRTT toolbox. At the beginning of each experiment day, it's advisable to repeat the three steps mentioned above to ensure that the hardware and experiment control scripts are configured correctly.

Finally, the current limitation of the toolbox is its implementation to one exclusive programming platform, which unfortunately compromises its use as a complete open-source toolbox. Nevertheless, in principle, the toolbox should be easily adaptable to other programming languages such as Python since the course of one trial remains unchanged.

In comparison to existing alternative methods, the 5-CSRTT toolbox introduced here allows the implementation of the standard 5-CSRTT paradigm and modifications of it such as defining a set time window for perseverative answers or introducing distracting or reinforcing stimuli like speakers or blinking lights. In addition to being easy to use and highly adaptable, the apparatus is low-cost and can be easily replicated, and it incentivizes research using non-rodent animal models.

## Disclosures

The authors declare that they have no competing financial interests.

## Acknowledgments

This work is supported by DFG WE 5469/3-1.

## References

1. Bari, A., Dalley, J. W., Robbins, T. W. The application of the 5-choice serial reaction time task for the assessment of visual attentional processes and impulse control in rats. *Nature Protocols*. **3** (5), 759-767 (2008).
2. Asinof, S. K., Paine, T. A. The 5-choice serial reaction time task: a task of attention and impulse control for rodents. *Journal of Visualized Experiments: JoVE*. (90), e51574 (2014).
3. Higgins, G. A., Silenieks, L. B. Rodent Test of Attention and Impulsivity: The 5-Choice Serial Reaction Time Task. *Current Protocols in Pharmacology*. **78** (1), 5.49.1-5.49.34 (2017).
4. Humby, T., Wilkinson, L., Dawson, G. Assaying aspects of attention and impulse control in mice using the 5-choice serial reaction time task. *Current Protocols in Neuroscience*. **31** (1), 8.5H.1-8.5H.15 (2005).
5. Robbins, T. The 5-choice serial reaction time task: behavioural pharmacology and functional neurochemistry. *Psychopharmacology*. **163** (3-4), 362-380 (2002).
6. Amitai, N., Markou, A. Disruption of performance in the five-choice serial reaction time task induced by administration of N-methyl-D-aspartate receptor

- antagonists: relevance to cognitive dysfunction in schizophrenia. *Biological Psychiatry*. **68** (1), 5-16 (2010).
7. Fizez, J., Cassel, J. C., Kelche, C., Meunier, H. A review of the 5-Choice Serial Reaction Time (5-CSRT) task in different vertebrate models. *Neuroscience & Biobehavioral Reviews*. **71**, 135-153 (2016).
  8. Spinelli, S. et al. Performance of the marmoset monkey on computerized tasks of attention and working memory. *Cognitive Brain Research*. **19** (2), 123-137 (2004).
  9. Parker, M. O. et al. Development and automation of a test of impulse control in zebrafish. *Frontiers in Systems Neuroscience*. **7**, 65 (2013).
  10. Birtalan, E., Bánhidi A., Sanders J. I., Balázsfi D., Hangya B. Efficient training of mice on the 5-choice serial reaction time task in an automated rodent training system. *Scientific Reports*. **10** (1), 1-8 (2020).
  11. Fischer, B., Wegener, D. Emphasizing the "positive" in positive reinforcement: using nonbinary rewarding for training monkeys on cognitive tasks. *Journal of Neurophysiology*. **120** (1), 115-128 (2018).