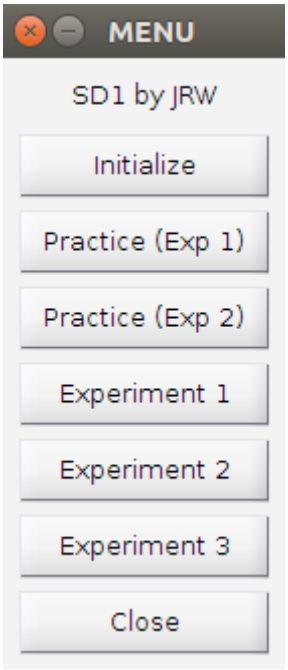
STOPDEVAL MANUAL

Jan R. Wessel, June 2015, jan.r.wessel@gmail.com

Table of Contents

1. Installation & startup

If you can read this document, you have already unzipped the file. Just change your current Matlab path into this directory, and you can start the program by executing the `SD1.m` file. This will bring up the main menu:

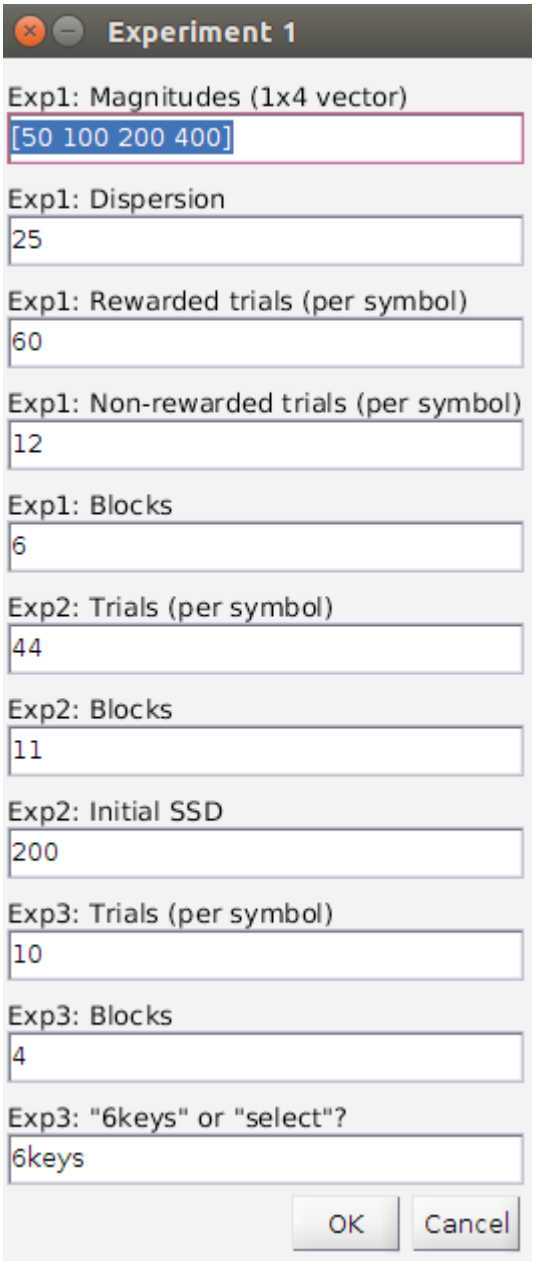


In order to run a participant, you will mostly work your way down this list of options.

2. Initialization

Click initialize. This will bring up the subject information mask:

Enter the subject information. Next, you will see the task options:



These options should be mainly straightforward if you are familiar with the Wessel et al., (2014) paper. Exp. 1 refers to the initial learning experiment, Exp. 2 to the treatment/stop-signal experiment, and Exp. 3 refers to the auction experiment.

2.1. Exp. 1 Options

2.1.1. Magnitudes

These are the values (in cents) that will be associated with the symbols. I have only run this experiment with these values in the past, but if you want to try different ones, you can change them here.
HOWEVER, you need to be careful to adjust the bids in the auction experiment so they still

cover your magnitude space. With the standard settings, possible bids in the auction range from 34c to 324c, which covers the space of expected value for the 50c-400c magnitude range [which, at a p(reward) of .8, results in expected values between 40c and 320c].
If you adjust the magnitudes to exceed this range (either by changing the magnitudes in this field, or the proportion of rewarded trials by adjusting the relative number of rewarded and non-rewarded trials [see below]), you need to adjust the possible bids.
This would have to be done within the code.
Currently, the bidding space for the 6 possible bids is defined in line 30 of the function `SD1p3sequence.m`, as

```
[start+level*scaling-1:...
start+level* scaling -1:...
(start+level* scaling -1)*stepsize]
```

The values for `start`, `scaling`, and `stepsize` are set in the function SD1settings.m, and are 30, 5, and 6, respectively. `Level` refers to each of the four values in the magnitude vector defined in this field. If you change the magnitudes in a way that exceed the range of 34c to 324c, make sure to adjust the bids by increasing / decreasing the `stepsize` parameter.

2.1.2. Dispersion

This is the range of the uniform distribution of values around the value means defined above.

2.1.3. {Rewarded; Non-Rewarded} trials (per symbol)

Self-explanatory. We use a ratio of 8:2 for rewarded vs. non-rewarded ($0.00). Changing these values can change this relation.

2.1.4. Blocks

Number of blocks into which the first experiment is going to be divided. Please make sure that the number of trials, given by

```
(Rewarded Symbols + Non-rewarded symbols) * length(magnitudes)*2
```

divided by the number of blocks defined here results in an integer value.

2.2. Exp. 2 Options

2.2.1. Trials (per symbol)

Number of trials per symbol. Please ensure that this number, multiplied by .75 (which is the probability of a stop-signal) results in an integer value.

2.2.2. Blocks

Number of blocks into which the second experiment is going to be divided. Please make sure that the number of trials, given by

```
Trials * length(magnitudes)*2
```

divided by the number of blocks defined here results in an integer value.

2.2.3. Initial SSD

Initial stop-signal delay for the tracking algorithm.

2.3. Exp. 3 Options

2.3.1. Trials (per symbol)

Number of trials per symbol. I would highly recommend to leave this at a number that is a multiple of 5. As a matter of fact, I have not tried any number other than 10. Using 10 presents each array of possible bids twice.

2.3.2. Blocks

Number of blocks into which the third experiment is going to be divided. Please make sure that the number of trials, given by

```
Trials * length(magnitudes)*2
```

divided by the number of blocks defined here results in an integer value.

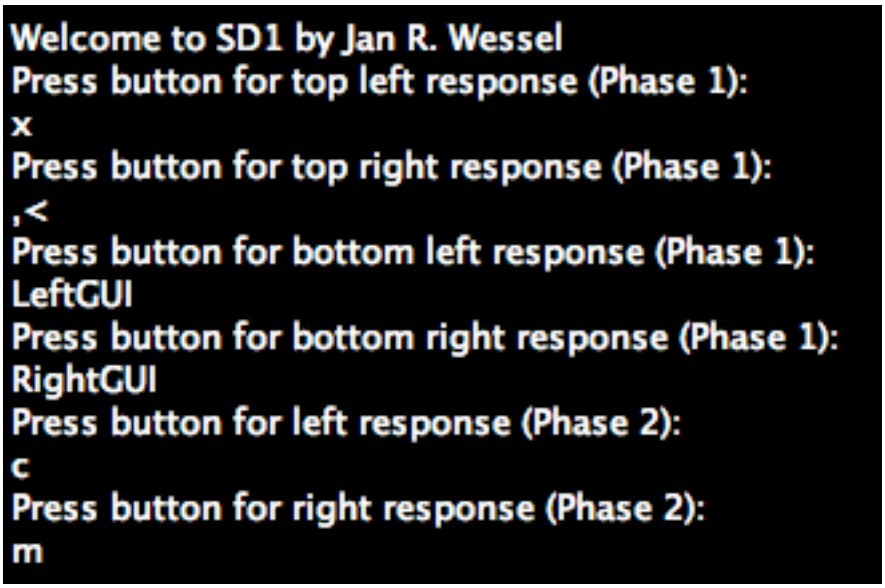
2.3.3. "6keys" or "select"

This allows toggling back between two different input modes for the bids. 6keys is the standard procedure used in Wessel et al., 2014, and defines the keys d,f,g,h,j, and k as selector keys for the six possible bids in the auction (these keys can be manually edited in the function `SD1settings.m`).

However, since some environment don't allow for the usage of a full-size QWERT style keyboard (e.g., fMRI), there is an alternative mode called 'select'. In this mode, one of the bids is going to be randomly highlighted in red at the beginning of the trial, which can then be moved around from left to right using the two keys that used to be bottom left and bottom right in Experiment #1 (the initial learning / reaction time task). The appropriate selection can then be confirmed by the subject by pressing one of the two upper keys from Experiment #1. This way, the entire task can be done on a four-button response device. In order to select that option, replace the string '6keys' in this field by 'select'.

2.4. Define response buttons

On the command line, you should now be prompted to define the response button by pressing them when prompted:



After doing this, the program should redirect you back to the main menu.

3. Instructions for Experiments 1 and 2

Apply the task instructions as described in Wessel et al., 2014. Instruct participants on Experiments 1 and 2 first, and tell them you will explain the third Experiment once they are done with the first two, as it is much more complicated.

3.1. Experiment 1

I usually instruct participants as follows:
“Experiment 1 is a reaction time task. On each trial, the screen is going to be divided into four quadrants by a big cross, and at some point, a shape is going to appear in one of the four quadrants. You will have four buttons to use in this experiment, one for each quadrant, and your task is to press that button as fast and accurately as possible. This is especially important since you are going to be rewarded with real money based on your task performance. After each response, the computer immediately tells you how much money you made on that trial, based on how fast and accurate you were. Now, because you are going to be doing this experiment for XX minutes and several {dozens/hundreds} of individual trials, we are not going to give you the entire amount you made on each trial, but instead, the computer will pick 5 trials at the end of the experiment, and we are going to pay you the money you made on those 5 trials. So, obviously, you want to do your best on each individual trial to maximize your potential reward.
You will do XX blocks of XX minutes each, for a total of XX minutes. You can take a break after each block, and the computer will give you an aggregate of fast and accurate you were in that particular block.”

3.2. Experiment 2

“Experiment 2 is similar to Experiment 1, but it has an additional component that makes it more challenging, but also a bit more interesting. On each trial, the screen is going to be divided into two halves by a horizontal line, and shapes are going to appear to the left or right of that line. You will use two buttons to respond to these stimuli as fast and accurately as possible, unless you hear a brief ‘beep’ tone after the shape appears. This tone is the so-called ‘stop-signal’, and it is called ‘stop-signal’ because it signals you to try to abort your response to the shape that appeared on the screen, and in fact, to try to not press any button whatsoever on this trial. Now, sometimes it is going to be pretty easy to do this, especially if, for example, the tone occurs at the same time as the shape, which will make it really easy to stop your response. However, sometimes the tone comes so late into your response that you will not be able to cancel your response, even though you will hear the stop-signal. It is completely normal that this will happen, and you will not be able to stop yourself every time you hear the stop-signal. It is important to realize that it is equally important to try to respond quickly when there is no stop-signal, and to stop successfully when there is a stop-signal.
You will do this experiment for XX blocks of XX minutes each, and we will provide you with detailed feedback about your performance after each of the blocks.

4. Practice

4.1. Experiment 1

By selecting ‘Practice (Exp. 1)’, you start a brief practice session (one block of seven trials) for the learning experiment, in which only a red rectangle is used as a stimulus shape (neither red as a color, nor a rectangle as a shape are used in the actual experiment), and the reward is always $1.00. This option can be used to familiarize the participant with the task mechanics for Experiment 1 (you should remind the participants that in the actual experiment, the rewards should be contingent on their reaction time and accuracy).

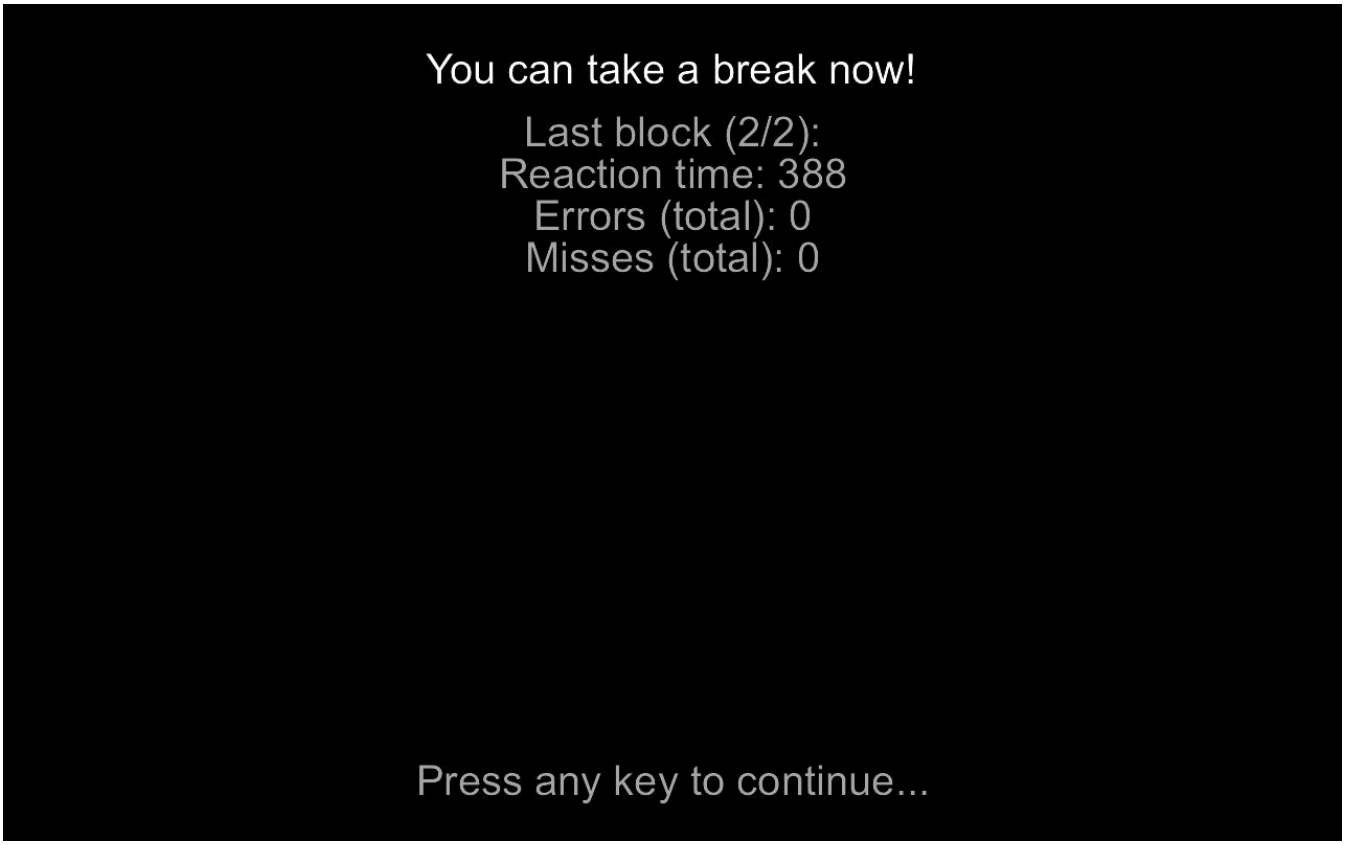
4.2. Experiment 2

By selecting ‘Practice (Exp. 2)’, you start a brief practice session (one block of ten trials, including four with stop-signals) for the stopping experiment, in which only a red rectangle is used as a stimulus shape (neither red as a color, nor a rectangle as a shape are used in the actual experiment),

5. Main Experiment

5.1. Experiment 1

Start Experiment 1 by selecting it from the main menu. After each block of the experiment, the participants will see the following screen:



This screen informs them of which block they just finished, what their RT was, as well as the number of errors and misses in the entire Experiment until this point. Pressing a key will continue the experiment. Once the last block is finished, the Experiment should terminate and the command line should come back up.

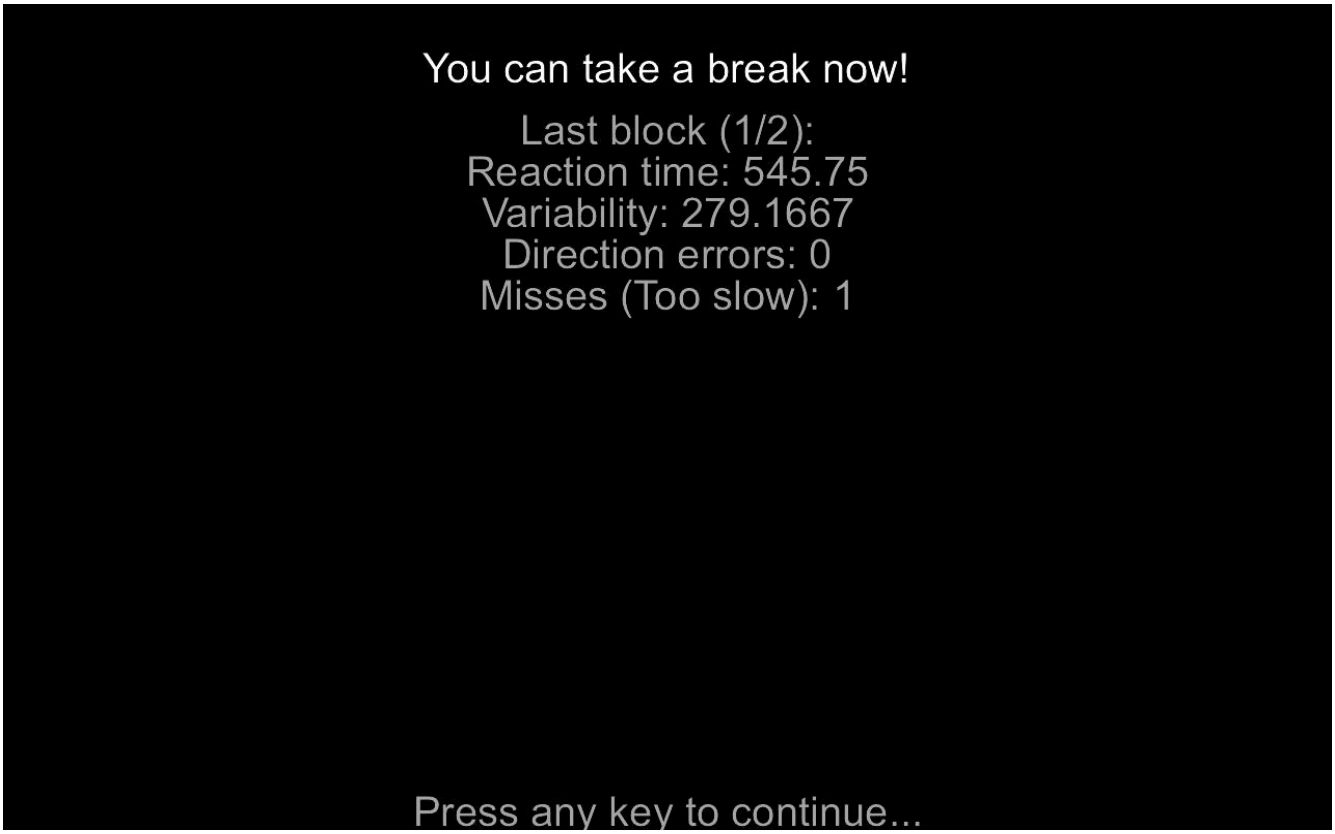
5.2. Experiment 2

5.2.1. Start

Start Experiment 2 by bringing up the main menu (execute `SD1.m`) and selecting Experiment 2. The Experiment will prompt you to select the Experiment 1 file, so please select the file '`SD1_Sxxx_Day-Month-Year_p1.mat`'.

5.2.2. Feedback screen

After each block, participants will see this feedback screen:



I am assuming that the reader is somewhat familiar with the mechanics of the stop-signal task. If not, please read the relevant literature before running this experiment. The stop-signal task is relatively involved for the experimenter, in that participants' behavior must be closely monitored. In a stop-signal with SSD tracking like the one used here, one generally wants p(inhibit), i.e., the success rate of stopping, to be around .5; however, this needs to be achieved without the participant knowing it (if they knew that they only succeed on stopping in 50% of cases regardless of what they do, they might lose motivation to properly stop).

In order to ensure that participants don't favor stopping over going (or vice versa), I use three parameters, two out of which are 'hidden' in the above display, and one (GoRT), which is openly available (Reaction time). The two hidden variables are p(inhibit), and mean SSD. P(inhibit) can be derived from the decimal numbers of the RT measurement. In the above case, stopping was successful on 75% of trials in the previous block. Mean SSD can be derived from the measurement of RT variability, which is in fact the mean SSD (i.e., mean SSD in the above example is 279.1667ms).

Generally, I attempt to keep participants within the following parameters, in order of preference:

1. Mean SSD > 100ms
2. GoRT between 400ms and 650ms
3. P(inhibit) between 40 and 60ms.

P(inhibit) is the least important to me, since the SSD tracking algorithm should automatically rectify deviations from this range over time, provided that the GoRT doesn't change.

5.2.3. Block feedback

Based on the above screen, I give participants feedback. As long as SSD is > 100ms, I just make sure that GoRT is within the range provided above.
If participants become too fast (which rarely happens), I tell them:
"Great Job – your reaction time on trials without stop-signals is very fast. However, do remember that you also want to try to stop when there is a stop-signal. So maybe respond a little bit more cautiously in the next block."
If participants become too slow (which happens regularly, especially when participants try to 'wait out' the stop-signal), I tell them:
"Great Job – you are doing a great job at stopping, but do remember that you also want to react quickly on trials on which there is no stop-signal. So maybe respond a little bit quicker in the next block."
This is the crucial part of the stop-signal task, the experimenter has to make sure that the right balance between going quickly and stopping successfully is struck.
Once the experiment is done, the command line should come back up.

5.3. Experiment 3

5.3.1. Instruction

At this point, I hand the participant the instruction sheet for the auction procedure. There are too different ones, based on which task mechanics (6keys or select) have been selected. They can be found in the SD1/manual directory.
After they have read through it carefully, I make sure the instructions are fully understood, and start up the experiment.

5.3.2. Start

Start Experiment 3 by bringing up the main menu (execute `SD1.m`) and selecting Experiment 3. The Experiment will prompt you to select the Experiment 2 file, so please select the file '`SD1_Sxxx_Day-Month-Year_p2.mat`'.

6. Post-experimental procedure

6.1. Probe of explicit knowledge

I usually like to do a systematic probe of the amount of explicit awareness of the two task contingencies participants have.

6.1.1. Learning contingency

Some participants will realize that their rewards in Experiment 1 do not depend on their own performance, but follow a schedule. These people tend to have a stronger devalution effect (Wessel et al., submitted). I probe this by asking an open question, then a more

specific question, and finally, by giving away the fact that there was a contingency and asking whether the participants could guess it. I usually use this sequence of questions:

1. Did you notice anything in the reaction time experiment?
2. Did you think you were better for some symbols than for others?
3. Some symbols were worth more than others, could you try to name any symbols you thought had particularly low or high values?

6.1.2. Stopping contingency

I also probe their awareness of the stopping contingency. While it might be too conservative or even unnecessary to do this, I like to exclude participants that develop an explicit awareness of the stopping contingency, just to be on the safe side with regards to potential task demand characteristics on the participants' side. I do this similarly to the above, e.g.:

1. Did you notice anything in the stop-signal experiment?
2. Did you think some symbols were paired with stopping more often than others?
3. Some symbols were paired with stopping more often. Could you try to name them?

6.2. Reward payout

The randomly selected rewards from Experiment 1 can be found in the settings file (see Section 7). For Experiment 3, I usually pay out a flat rate reward (but obviously still instruct them that their bids will translate into higher / lower rewards during the instruction) out of simplicity. Recently, I have also resorted to doing this in Experiment 1, since their behavior actually has no bearing on the rewards, and a very good (fast/accurate) subject could randomly get much lower rewards than a very bad participant. Obviously, these things are up to you (and your IRB).

7. Log files and data storage

At the end of the experiment, you will end up with one single datafile in the 'out' folder of your SD1 directory, called `SD1_Sxxx_Day-Month-Year.mat`. The datafile contains the following variables.

7.1. The `data` structure

Contains demographic data and subject number.

7.2. The `output` structure

Structure that contains the subject's performance / response parameters for each individual experiment. For Experiment 1, it contains the combined error and miss rate, as well as the RT overall and for all individual shapes. For Experiment 2, it contains the trial numbers for each trial type, the miss rate, error rate, and stopping success rate (overall and for each shape individually), RT overall and for all individual shapes, and stopping-related

parameters (SSRT and mean SSD). For Experiment 3, it contains the overall bidding value for stopping and non-stopping symbols (across all shapes), as well the mean individual bidding level, mean bidding amount, bidding level variance, and RT for each individual shape.

7.3. The `settings` structure

Structure that contains a log of all the settings used for this participant, as well as the reward from the learning phase (randomly picked from five trials, found under `settings.reward`).

7.4. The `trialseq_p*` matrices

Contain the raw data for each individual file, which underlie the `output` structure. Column indices can be found in the `SD1columns.m` file, and are as follows:

Column 1: Raw reward magnitude [Phase 1]
Column 2: Shape (1 = low, no stop; 2 = mid low, no stop; 3 = mid high, no stop…)
Column 3: Stepwise reward magnitude (1 - 4)
Column 4: Generally paired with stopping or not?
Column 5: Shape (triangle, flipped triangle, square, diamond, cross, hexagon, I, circle)
Column 6: Color (green, blue, cyan, magenta, yellow, orange, white, gray)
Column 7: Quadrant (1 = top L; 2 = top R; 3 = bottom R; 4 = bottom L; 5 = L; 6 = R)
Column 8: Response (1 = top L; 2 = top R; 3 = bottom R; 4 = bottom L; 5 = L; 6 = R)
Column 9: Reaction Time
Column 10: Block
Column 11: Number of trial presentations (errors and misses are replayed in Phase 1)
Column 12: Stop Signal? [Phase 2]
Column 13: Left SSD [Phase 2]
Column 14: Right SSD [Phase 2]
Column 15: Accuracy (1 = Correct; 2 = Error; 3 = Failed stop; 4 = Successful stop; 5 = Miss)
Column 16: Bidding amount chosen [Phase 3]
Column 17: Bidding level chosen [Phase 3]
Col. 18-23: Possible bids (in order on screen) [Phase 3]