Resource bounds on mental simulations (Supplementary Materials)

YingQiao Wang [1], Tomer D. Ullman[1,*]

*Corresponding author. Email: tullman@fas.harvard.edu

Contents

1 Overview

We hypothesized that when using intuitive physical reasoning, people can make use of a mental simulation. However, this mental simulation is not an fully accurate copy of reality: it uses approximations, and has capacity limitations. Specifically, we hypothesized that there are capacity limitations that can be captured by a limited budget of 'particles' in a particle-based simulation. We examined this hypothesis using 4 main novel pre-registered experiments that examine reasoning about containers, volume, and fluids.

We predicted that people would be competent when reasoning about the behavior of liquids over a short period of time, but that after a certain period of time, people's behavior would change when they hit a capacity limit.

In Experiments 1 and 2, we established baseline performance on two tasks that examine reasoning about liquids in dynamic scenes. We asked people to judge when different containers will be fully filled by a stream of water (Experiment 1), and how filled a container is when different amounts of water are poured into it (Experiment 2). In Experiment 3, we examined people's estimation of container volume in static scenes, and found they have systematic biases in their estimations. In Experiment 4, we increase the presentation time of stimuli from Experiment 1, and find a 'switch point' after which people's behavior changes. We additionally had people perform a digit-span task, and found that people with larger digit spans have a later switch-point.

A mental simulation model with limited particles can account for both the successes of Experiment 1 and 2, as well as the change in behavior past the switch point in Experiment 4. This model suggests people run out of a limited budget of "particles", after which they need to recycle existing ones and re-scale them.

In this supplementary, we provide additional information on the statistical methods we used, the creation of the stimuli, and the computational cognitive model. We also detail supplementary experiments, S1 and S2, that examined in more specifically the relation between time estimation and cognitive capacity (S1) and also probe the relationship between the flow rate of the liquid, and the location of the switch point (S2).

2 Sampling Methods, Hypothesis Testing, and Model Comparison Metrics

When assessing people's behavior in Experiments 1, 2, and 4, we compared their behavior to several hierarchical statistical models (see more details below). For inference, we used No U-Turn Sampler (NUTS), which is a special type of Hamiltonian Monte Carlo (HMC) algorithm, to sample the posterior distribution of the parameters in all the hierarchical linear models we constructed in the analysis.

The NUTS algorithm used 8 chains. Within each chain, 20000 samples were drawn, including 10000 samples as burn-in samples for tuning purpose. The target acceptance probability was 0.95.

For our analysis on the effect of shape in Experiments 1 and 2, we pre-selected 95% as the interval range for our Highest Density Interval (HDI). Our reasoning in selecting this range was in line with established practice [6], and the 95% interval resembles the 95% confidence interval in frequentist hypothesis testing. In using such an HDI parameter, one must take care to consider computational instability [5]. To tackle such potential instabilities, it is recommended to take a sufficiently large sample size, usually around 10,000 samples. Our analysis sampled 10,000 samples across 8 chains, and surpassed the sample size suggested by the experiment in [5].

One of our analyses aimed to test whether the shape of a container affected people's performance. A negative claim (that it does not) is valid under a Bayesian analysis framework. That is, under the logic of Bayesian hypothesis testing, it is possible to use the absence of specific evidence as evidence in favor of the null hypothesis (that container shape does not have an effect). We stress that whether shape actually had an effect in E1, E2, and E4 is not counter to the mental simulation hypothesis, and that it is sufficient for our purposes that whatever the effect was, it was different in E3.

Our model comparison was based on Leave-One-Out Cross Validation. The Leave-One-Out Cross Validation results were estimated based on Pareto-Smoothed Importance Sampling (PSIS) [10]. Given the Leave-One-Out Cross Validation results, stacking was used to calculate the model weights [11].

The stacking process was as follows: Given a dataset with entry (x_i, y_i), where $i \in 1, ...n$, we fitted model $M_1 ....M_k$. We assumed that each model M_k has a parametric form $\hat{y_k} = f_k(x|\theta_k)$, where θ_k is the model parameter. The Leave-One-Out Cross Validation predictor $f_k^{-i} = E[y_i|\hat{\theta}_{k,y-i}, M_k]$ was estimated using PSIS for each model k and each data point i. The model weight was obtained by minimizing the Leave-One-Out (LOO) mean square error:

$$\hat{w} = \arg\min_{w} \sum_{i=1}^{n} (y_i - \sum_{k} w_k \hat{f_k}^{-i}(x_i))^2 \tag{1}$$

All the sampling and model comparisons were completed using built-in methods in PyMC, a Bayesian Probabilistic Programming Language based on Python.

3 Stimuli Specification

We used the Unity 3D game engine for all stimuli generation `https://unity3d.com`. The particle-based fluid simulation engine is provided via the Obi Fluid package, which can be purchased as an add-on package in the Unity Asset Store.

The particle-based fluid simulation engine has several parameters, including the target amount of particles emitted, flow rate, particle size, particle lifespan, and emitter disk radius. Other parameters include the physical properties of the simulated fluid, and in our case we used the rest density, buoyancy, surface tension, and viscosity of water. All of these parameters were held consistent throughout all stimuli.

According to the output of the Unity game engine, assuming the standard-sized water particle has size 1 (in arbitrary units), the water flow rate is constantly 209 standard-sized water particles per second.

In Experiments 1 and 4, the *volume* (in standard-sized water particles) of the containers in each trial was $209 * TTF_{groundTruth}$. For example, if the ground-truth 'Time-to-Fill' of the container was 1 second, then the corresponding container had a volume of 209 standard-sized water particles. So, the volume of the containers (in units of standard water particles) in Experiment 1 was 209, 418, 627, 836, 1045, 1254, and 1463. In Experiment 4, the volume of the containers (in units of standard particles) was 209, 418, 627, 836, 1045, 1254, 1463, 1672, 2090, 2717, 3344, 3971, 4598.

For Experiment 2, The volume of the liquid within the containers in each trial was $209 * duration$. So, the volume that the liquid took up within the containers (in units of standard water particles) was be 418, 627, 836, 1045, and 1254.

For Experiment 3, we used containers of 3 different volumes: 698, 1047, and 3490 (in units of standard water particles). Using this we created four different volume ratios: 698:698, 698:1047, 1047:3490, and 698:3490, corresponding to 1:1, 1:1.5, 1:3, and 1:5.

4 Experiment 1: Analysis of the effect of shape

As mentioned above and in the main paper, we were interested in examining the possible effect of object shape on people's performance in the different experiments, but emphasize that there is no a-priori reason to think object shape *won't* matter for a simulation model.

The hierarchical linear models fitted on the data for different shapes are shown in figure 1. We emphasize that the overlapping HDI for the hierarchical linear model fitted on the wide cup shape is not visually obvious in the figure. The reason for this could be traced to the inferred posterior distribution for the parameters of the model. Although the HDI for both slope and intercept parameters overlapped for both the wide and regular cup, small differences in the intercept are magnified in a way that makes the visualization such that the lines appear non-overlapping visually, but that is not the basis for the statistical analysis.

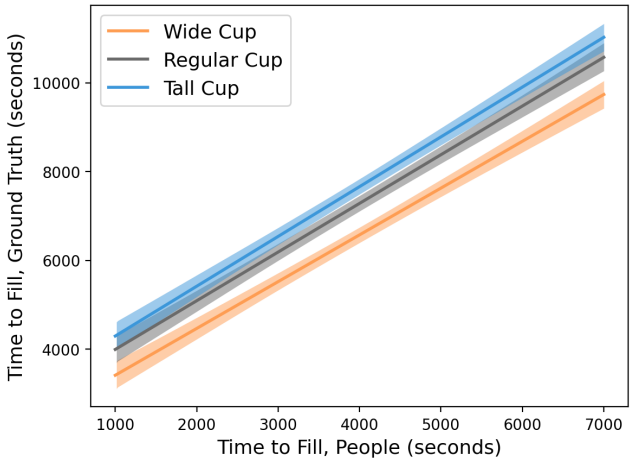

Figure 1: The hierarchical linear model fitting for cups with different shape. The hierarchical linear model fitting for the wide cup is indicated in orange. The hierarchical linear model fitting for the regular cup is indicated in gray. The hierarchical linear model fitting for the tall cup is indicated in blue. Note that the visualization is misleading in making the lines appear separate, while the HDIs over the parameters of the hierarchical linear models (the basis for the statistical analysis) are overlapping.

5 Experiment 2: Analysis of the effect of shape

As in Experiment 1, we analyzed the effect of the cup shape on the data from Experiment 2. The hierarchical linear models fitted on the data for different shape are shown in figure 2,

6 Experiment 3: The Effect of Viewing Angle on the Estimation of Volume

In Experiment 3, we examined whether particular shape biases will appear when people are estimating volumes statically. Previous work suggests that people may exhibit an 'elongation bias' when estimating volume [8, 9], but more recent work [1, 2] suggests this bias is more complicated than previously described. This more recent work finds that different view point angles provide different visual cues, which causes the elongation bias to vary and switch. Our stimuli included showing people the volumes from several viewing angles, leaving it open a-priori which bias would emerge.

The visual bias we empirically found in Experiment 3 was a clear bias towards wider containers, which is in line with [2]. Our current stance is that the static estimation of volume is a complex process with several biases that interact with object size, shape, and viewing angle. Our goal in Experiment 3 (and in the work overall) was *not* to establish a comprehensive set of features and biases used in static volume estima-

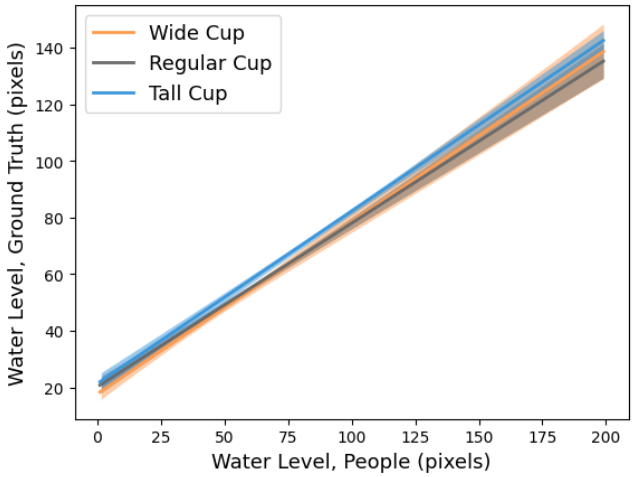

Figure 2: The hierarchical linear model fitting for cups with different shape. The hierarchical linear model fitting for the wide cup is indicated in orange. The hierarchical linear model fitting for the regular cup is indicated in gray. The hierarchical linear model fitting for the tall cup is indicated in blue. The three hierarchical linear models have overlapping HDI.

tion. Rather, it was sufficient for our purposes that within the limited stimuli used in the experiment, whatever features are used for static estimation differ from dynamic simulation.

7 Experiment 4: An Internal Reliability Analysis for the Digit Span Forward Task

The Digit-Span Forward Task is an established measure that shows high internal consistency, and test-retest ability. For example, [3] demonstrated that the digit-span forward task has an internal reliability of around [0.7 - 0.9] using the split-half method, and in [4], the internal reliability for the digit-span forward task was found to be 0.89 using the split-half method. While previous work has demonstrated the high internal reliability measure for the digit-span forward task, we still implemented a test-retest reliability measure for our digit-span forward task.

7.1 Analysis Details and Results

We implemented the ”test-retest reliability: intraclass correlation coefficients” metric detailed in [7]. The test-retest reliability is most suitable for our task design, as we asked people to complete the digit-span task twice. We found that the intraclass correlation coefficient for our digit-span forward task is 0.66, with 95% CI [0.55 - 0.76],

somewhat lower than the previous studies, but overall in keeping with test-retest reliability.

8 Experiment 4 Additional Analysis: Treating the digit span as an ordinal variable, rather than using a median split

In our main analysis of Experiment 4, we split the participants into two groups according to their performance in the working memory tasks (digit-span forward task), and inferred the switch point position for each corresponding groups. This is equivalent to treating performance in the digit span task as a *binary* variable. Here, we detail an additional analysis that treats performance in the digit span tasks as an *ordinal* variable, and infer their switch point position correspondingly. We note that this analysis was not pre-registered, and we thank a Reviewer for suggesting it.

8.1 Analysis Details

In our new analysis, the participants were separated by their digit span score. We note that performance was not equally distributed, nor did we expected it to be. For the 5-score group, there are 5 participants, meaning that there are 5 data points. For the 6-score group, there are 20 participants / data-points. For the 7-score group, there are 47 participants / data-points. For the 8-score group, there are 40 participants / data-points. For the 9-score group, there are 66 participants / data-points.

We inferred a participant's switch point position conditioned on the digit span score, using a hierarchical linear model fitted on the dataset corresponding to participants with the score of 5, 6, 7, 8, 9. The linear model fit is inherited from Experiment 4, but the prior setting changed. Specifically:

Priors for Parameters	
Parameters	Linear
a	Uniform(0,5000)
b	Uniform(0,2)
c	Uniform(-10,10)
$SwitchPoint$	Uniform(2000,19000)
ε	Uniform(0,4000))

The priors for the individual hyperprior parameters were the following:

Priors for Parameters	
Parameters	Linear
a_i	Normal(a,1250)
b_i	Normal(b, 0.24)
c_i	Normal(c, 0.24)
$SwitchPoint_i$	Normal($SwitchPoint$, 1000)
ε_i	Normal(ε, 1000)

After obtaining the mean inferred switch point position from the hierarchical model fitting on groups separated by digit span score, we fit a simple linear model to the inferred switch point position. The model is the following:

$$MeanInferredSwitchPointPosition = a + b * digitspanscore \quad (2)$$

We would like to highlight that the parameter of this model was estimated using maximum likelihood estimation, contrasting with our Bayesian approach used in the rest of our analysis. So, no prior and posterior distribution are involved.

8.2 Results

The mean inferred switch point position for the 5-score group was 5121ms, with 95% HDI [2000, 9420]. The mean inferred switch point position for the 6-score group was 6714ms, with 95% HDI [2013, 11124]. The mean inferred switch point position for the 7-score group was 8015ms, with 95% HDI [3001, 12979]. The mean inferred switch point position for the 8-score group was 10500, with 95% HDI [6906, 16333]. The mean inferred switch point position for the 9-score group was 9568ms, with 95% HDI [5967, 14257].

According to the model fitting outputs of the simple linear model fitted on the mean inferred switch point position, the maximum likelihood estimator for the intercept parameter a was -892.4, with 95% CI [-4991,3206]. The maximum likelihood estimator for the slope parameter b was 1268, with 95% CI [694,1842].

In other words: there is a significant increase in the switch point (further along in time), as the digit-span increases.

We would like to emphasize that the unequal distribution of people across the groups makes a point-by-point comparison harder. But, both analyses (median-split and ordinal analysis) point in the same direction: participants with a higher digit span score show a later switch point position.

9 The Effect of Choice of Priors

In Bayesian statistics, the choice of prior could be critical if the data is not large enough to dominate the inference process. In our experiment, we adapted the Bayesian inference framework for our hierarchical linear model analysis, and so it is critical to either state the reasoning behind the prior choice, and also to experimentally test if our inference process is data-dominated. We would like to argue that our inference framework for our hierarchical linear model analyses are data-dominated, and all the priors provided very little to no information towards our inferred posterior distribution.

We tested various prior range choices for the intercept and slope in our hierarchical models. For the intercept terms in the hierarchical model with linear, square root, and logarithmic functional form, we used [0,4000] for the intercept for the hierarchical linear model with linear functional form, but [-1000,1000] for the hierarchical linear model with square root and logarithmic functional form. As Figure 4 shows, if the prior is specified within a proper rage, then the posterior distribution will approximate the true posterior distribution, in this case is normal distribution. We also tried to use

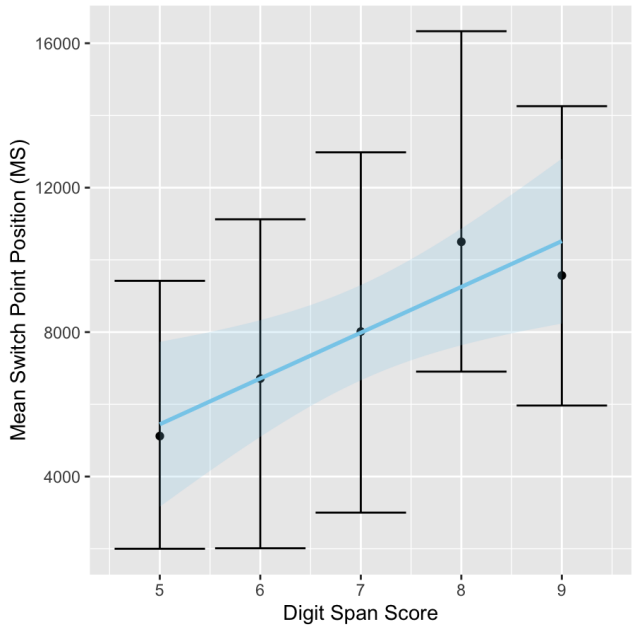

Figure 3: The inferred switch point position's mean (black dots) and 95% confidence intervals (black lines), along with the linear model fit (blue line) and 95% confidence interval (shaded area) of inferred switch point position over people's digit span forward task score.

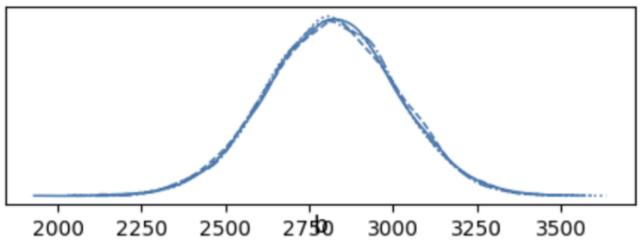

Figure 4: Density Plot of the Posterior Distribution of the Intercept Parameter for the Linear Model in Experiment 1. The x-axis is the Range of the Intercept Parameter.

[0,4000] for the intercept for the hierarchical linear model with square root and logarithmic functional form, but as Figures 5 and 6 show, the prior range was not appropriate; the data dominated the inference process and the inferred posterior distribution was incomplete, indicating that this prior was specified with wrong range.

Having demonstrated that choosing priors with an inappropriate range could be detected by the inference algorithm, thus causing data to dominate and report that priors' ranges are inappropriate, our strategy changed to choosing extremely uninformative priors with huge ranges, even though such a range would be way wider than the true

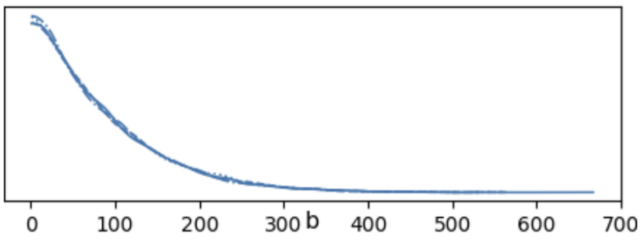

Figure 5: Density Plot of the Posterior Distribution of the Intercept Parameter for the Square Root Model in Experiment 1. The x-axis is the Range of the Intercept Parameter.

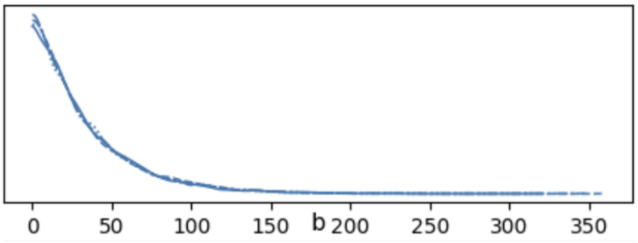

Figure 6: Density Plot of the Posterior Distribution of the Intercept Parameter for the Logarithmic Model in Experiment 1. The x-axis is the Range of the Intercept Parameter.

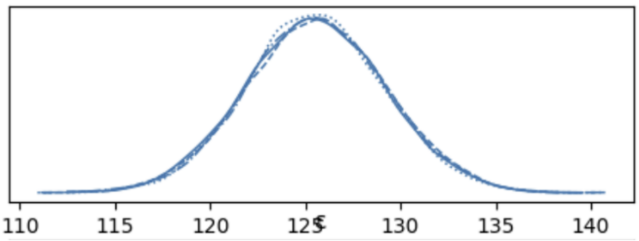

Figure 7: Density Plot of the Posterior Distribution for the Group Slope Parameter for the Square Root Model Under Prior (Uniform [0,150]) in Experiment 1. The x-axis is the Range of the Group Slope Parameter.

range. We tested the hierarchical linear model with square root functional form, since this model has relatively larger slope. We used [-150,150] and [0,150] for the slope parameter. As Figure 7 and 8 shows, two prior ranges, [-150,150] and [0,150], inferred nearly-identical posterior distribution for our slope parameter.

From the experimental results above, we determined that priors in our inferences are indeed uninformative. Although our priors are uninformative and posterior distributions are dominated by data, our experimental results showed that choosing a wide prior range, even ranges that are way wider than the true posterior distribution, would

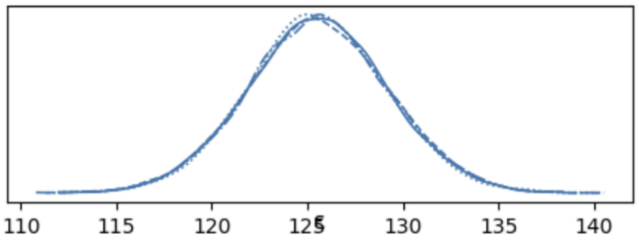

Figure 8: Density Plot of the Posterior Distribution for the Group Slope Parameter for the Square Root Model Under Prior (Uniform [-150,150]) in Experiment 1. The x-axis is the Range of the Group Slope Parameter.

be the appropriate strategy here for the prior ranges' selection.

10 Experiment 1, 2, 4: Additional Model Comparison Results

In our main paper, the model comparison involved the models with a specified functional form, like linear, square root, and logarithmic. The lack of freedom to let the algorithm decide the functional form is a potential weakness in our model comparison. So, we used a (post-hoc) analysis to consider a model with a free-exponent parameter. We thank a Reviewer for suggesting this strategy.

In Experiment 1, the free-exponent model was defined as:

$$TTF_{Human} = a + b * TTF^{r}_{groundTruth} + \varepsilon \tag{3}$$

where the priors for the parameters are shown as the following:

Priors for Parameters	
Parameters	Linear
a	Uniform(0,4000)
b	Uniform(0,100)
r	Uniform(-5,5)
ε	Uniform(0,7000))

The priors for the individual hyperprior parameters were the following:

Priors for Parameters	
Parameters	Linear
a_i	Normal(a,2000)
b_i	Normal(b, 0.6)
ε_i	Normal(ε, 1000)

The estimated hyperprior parameters (mean and 95% HDI) for the free-exponent model were: a = 2809.6, with 95% HDI [2428.0, 3195.9]; b = 1.1, with 95% HDI [0.92, 1.3]; r = 0.99, with 95% HDI [0.98, 1.01]; ε = 2661.5, with 95% HDI [771.8, 4262.9].

Since this range includes the linear model, we did not proceed further.

In Experiment 2, a free-exponent model was defined as:

$$WL_{Human} = a + b * WL^{r}_{groundTruth} + \varepsilon \tag{4}$$

where the priors for the parameters are shown as the following:

Priors for Parameters	
Parameters	Linear
a	Uniform(0,1000)
b	Uniform(0,100)
r	Uniform(-5,5)
ε	Uniform(0,100))

The priors for the individual hyperprior parameters were the following:

Priors for Parameters	
Parameters	Linear
a_i	Normal(a,4)
b_i	Normal(b, 0.2)
ε_i	Normal(ε, 10)

The estimated hyperprior parameters (mean and 95% HDI) for the free-exponent model: a = 23.5, with 95% HDI [21.4, 25.6]; b = 0.4, with 95% HDI [0.3, 0.5]; r = 1.1, with 95% HDI [1.06, 1.12]; ε = 21.2, with 95% HDI [3.2, 39.1].

Since the model is (just barely) not a linear model, an additional model comparison based on the Leave-One-Out Cross Validation were conducted. According to the model comparison result, MW(Linear) = 0.5; MW(Square Root) = 0.02; MW(Logarithmic) = 0.1; MW(Free Exponent) = 0.38.

In Experiment 4, a free-exponent model was defined as:

$$TTF_{Human} = a + b * TTF^{r}_{groundTruth} + \varepsilon \tag{5}$$

where the priors for the parameters are shown as the following:

Priors for Parameters	
Parameters	Linear
a	Uniform(0,5000)
b	Uniform(0,100)
r	Uniform(-5,5)
ε	Uniform(0,5000))

The priors for the individual hyperprior parameters were the following:

Priors for Parameters	
Parameters	Linear
a_i	Normal(a,1500)
b_i	Normal(b, 0.5)
ε_i	Normal(ε, 1000)

The estimated hyperprior parameters (mean and 95% HDI) for the free-exponent model were: a = 2961.2, with 95% HDI [2724.3, 3203.8]; b = 0.6, with 95% HDI [0.47, 0.63]; r = 0.97, with 95% HDI [0.96m 0.98]; ε = 2109.3, with 95% HDI [220.9, 3870.3].

Since the model (just barely) not a linear model, additional model comparison based on the Leave-One-Out Cross Validation was conducted. According to the comparison result, MW(Linear) = 0.0; MW(Square Root) = 0.0; MW(Logarithmic) = 0.0; MW(Linear&Linear) = 0.83; MW(Linear&Square Root) = 0.170; MW(Linear&Logarithmic) = 0.0, MW(Free Exponent) = 0.0.

In sum, a free-exponent model analysis confirms the selection of the linear model.

11 Experiment S1: General Time Estimation and Cognitive Capacity

Our main analyses for Experiment 4 showed (i) a switch point, and (ii) a connection between cognitive capacity (as measured in the digit span task) and bounded mental simulation resources (as measured in the position of the switch point in the fluid reasoning tasks). But, it is possible that there are more general relationships affecting these findings that have nothing to do with mental simulation per se. For example, it is possible that working memory limitations cause a switch point in time estimation regardless of simulation, or that people with more limited working memory have a different estimation of time independent of any simulation. To examine this, we conducted a new experiment, in which we examine the relationship between cognitive capacity and general time estimation. We note that this experiment was not part of our original pre-registration, and thank an anonymous Reviewer for suggesting it.

11.1 Experimental Design

In order to test the connection between cognitive capacity and time judgment in a way that mimics that overall set-up of our studies while not using mental simulation, we constructed a time-interval judgment task. In this task, participants are first presented with a time gap between two tones, of varying length T. Participants are next presented with a third tone, and asked to press the spacebar when they think the same time gap T had passed. To align with the Experiment 4, we used 1, 2, 3, 4, 5, 6, 7, 8, 10, 13, 16, 19, and 22 seconds as the time gap T. In addition, we used a working memory test, which was exactly the same as the one in Experiment 4.

We recruited 50 US-based participants, online. We excluded 9 participants based on our pre-registered exclusion criterion for Experiment 4. This left 41 participants for analysis (30 identified as women, 11 identified as men. The median age was 40, the mean age was 39.5, and the standard deviation was 13). The average completion time of the study was 22.5 minutes, with standard deviation of 6.2 minutes. The participants were paid at a rate of $12/hrs.

11.2 Analysis Details

We used a two-sample t-test and hierarchical linear models in our analysis. As in the main analysis of Experiment 4, we divided participants into two groups according to their performance on the digit span task.

For the two-sample t-test, we created a new metric $TimeDifference$, which is the difference between a participant's perceived time gap and the ground truth time gap. A Welch Two-Sample T-Test was then conducted on $TimeDifference$.

For the hierarchical linear model, we inherited exactly from Experiment 4, which we used hierarchical linear model with linear functional form, and hierarchical linear model with linear functional form and switch point for the analysis. Although the model setup is the same, please note that we used slightly different priors for the hyperprior parameters and individual hyperprior parameters, as noted below:

The priors for the hyperprior parameters of the hierarchical linear model with linear functional form without switch point were the following:

Priors for Parameters	
Parameters	Linear
a	Uniform(0,5000)
b	Uniform(0,2)
ε	Uniform(0,50000))

The priors for the hyperprior parameters of the hierarchical linear model with linear functional form without switch point were the following:

Priors for Parameters	
Parameters	Linear
a_i	Normal(a,500)
b_i	Normal(b, 0.1)
ε_i	Normal(ε, 1000)

The priors for the hyperprior parameters of the hierarchical linear model with linear functional form with switch point were the following:

Priors for Parameters	
Parameters	Linear
a	Uniform(0,5000)
b	Uniform(0,2)
c	Uniform(-10,10)
$SwitchPoint$	Uniform(2000,19000)
ε	Uniform(0,4000))

The priors for the hyperprior parameters of the hierarchical linear model with linear functional form with switch point were the following:

Priors for Parameters	
Parameters	Linear
a_i	Normal(a,1250)
b_i	Normal(b, 0.24)
c_i	Normal(c, 0.24)
$SwitchPoint_i$	Normal($SwitchPoint$, 2000)
ε_i	Normal(ε, 1000)

11.3 Results

A Welch Two-Sample T-test examining the metric $TimeDifference$ between the low-digit group and the high-digit group was not statistically significant ($t = 0.2039$, $df = 315.68$, $p = 0.83$). This indicates that on average, the time gap perceived by participants aligned with the ground truth, and that there was no relationship between cognitive capacity and the time gap perceived by the participant.

The estimated hyperprior parameters (mean and 95% HDI) for the model with linear functional form on low-digit group were: a = 1162.032, with 95% HDI [-235.0, 2557.0]; b = 0.909, with 95% HDI [0.774, 1.042]; ε = 4667.5, with 95% credible interval [2649.5, 6741.4].

The estimated hyperprior parameters (mean and 95% HDI) for the model with linear functional form and switch point on low-digit group were: a = 899.7, with 95% HDI [-437.0, 2247.5]; b = 0.887, with 95% HDI [0.455, 1.328]; c = 0.024, with 95% HDI [-0.448, 0.478]; $SwitchPoint$ = 4304.8, with 95% HDI [2000.1, 7865.3]; ε = 3312.7, with 95% credible interval [1360.9, 5312.9].

The estimated hyperprior parameters (mean and 95% HDI) for the model with linear functional form on high-digit group were: a = 743.7, with 95% HDI [-42.0, 1529.1]; b = 0.930, with 95% HDI [0.852, 1.008]; ε = 3860.2, with 95% credible interval [1865.5, 5852.1].

The estimated hyperprior parameters (mean and 95% HDI) for the model with linear functional form and switch point on high-digit group were: a = 881.1, with 95% HDI [-449.2, 2271.1]; b = 0.893, with 95% HDI [0.455, 1.328]; c = 0.016, with 95% HDI [-0.457, 0.474]; $SwitchPoint$ = 4254.9, with 95% HDI [2000.1, 7658.8]; ε = 3302.2, with 95% credible interval [1285.2, 5266.8].

Notice that the 95% credible interval for parameter c for all models with a switch point on low-digit and high-digit group contains 0, indicating that the slope modification determined by switch point position is not statistically significant. So, the switch point model was not be selected in both cases, indicating that the behavior for the participants in this task is consistent across all ground truth time gap T, and does not show the switch-point that we found for mental simulation.

We plotted the hierarchical linear models with linear functional form and without switch point position fitted on the low/high-digit groups 9. As can be seen, the overlap in between models indicates no relationship between cognitive resources (as estimated by the digit span task) and the time gap estimation task:

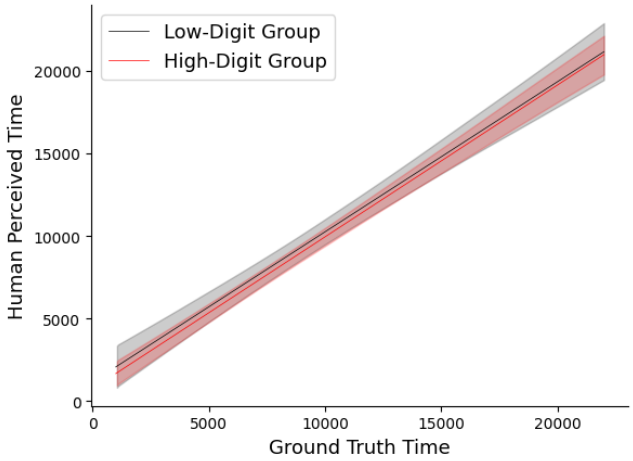

Figure 9: **Results of Experiment S1**The Hierarchical Linear Models Without Switch Point Fitted on the Low/High-Digit Groups, and the 95% HDI for both Models. The models overlap, indicating no effect of cognitive resources on the time estimation task.

12 Experiment S2: The Relationship Between Flow Rate and the Location of the Switch-point

In addition to ruling out some general effects of cognitive capacity independent of mental simulation, we examined more positive evidence for the link between cognitive capacity and mental simulation, by modifying the original flow rate of Experiment 4 and slowing it down. Under our model, this leads to less particles being used over the same span of time, and so should lead to particles 'running out' later in time, pushing the switch point later in time. We note that this experiment was not part of our original pre-registration, and thank an anonymous Reviewer for suggesting it.

12.1 Experimental Design

We adopted the exact same experimental and stimuli design as in Experiment 4. The only modification was that in every stimuli, the flow rate was half of that of the original study, resulting in the quantity $TTF_{GroundTruth}$ being doubled to 2, 4, 6, 8, 10, 12, 14, 16, 20, 26, 32, 38, 44 seconds.

We recruited 156 US-based participants, online. We excluded 27 participants based on our pre-registered exclusion criterion for experiment 4. This left 129 participants for analysis (90 identified as women, 38 identified as men, 1 person declined to answer. The median age was 40, the mean age was 41, and the standard deviation of age was 12.5).The average completion time of the study was 24.5 minutes, with standard deviation of 10 minutes. The participants were paid at a rate of $12/hrs.

12.2 Analysis Details

Inherited from Experiment 4, we used a hierarchical linear model with linear functional form and switch point for the analysis. Although the model setup is the same, we used slightly different priors for the hyperprior parameters and individual hyperprior parameters, as noted below:

The priors for the hyperprior parameters were the following:

Priors for Parameters	
Parameters	Linear
a	Uniform(0,20000)
b	Uniform(0,2)
c	Uniform(-10,10)
$SwitchPoint$	Uniform(4000,38000)
ε	Uniform(0,20000))

The priors for the individual hyperprior parameters were the following:

Priors for Parameters	
Parameters	Linear
a_i	Normal(a,2000)
b_i	Normal(b, 0.5)
c_i	Normal(c, 0.5)
$SwitchPoint_i$	Normal($SwitchPoint$, 3000)
ε_i	Normal(ε, 2000)

12.3 Results

The estimated hyperprior parameters (mean and 95% HDI) for the model with a linear functional form and switch point on all the participants were: a = 5768.1, with 95% HDI [4854.3, 6687.7]; b = 0.907, with 95% HDI [0.783, 1.037]; c = -0.665, with 95% HDI [-0.801, -0.529]; $SwitchPoint$ = 16624.9, with 95% HDI [14625.5, 18642.0]; ε = 5424.6, with 95% credible interval [1393.3, 9123.8].

These results suggest that after reducing the flow rate in half, the switch point was roughly doubled. More specifically, after reducing the flow rate by 0.5, the switch point was 1.8 times the original switch point, moving from 9.7 seconds to 16.6 seconds). So, in line with our general framework, we were able to obtain more positive evidence for the link between the simulation and the working memory span by controlling for the time within the experiment.

While not part of the main analysis, we note that for completeness we again split the participant by their working memory span to examine varying switch points. Here, we did not find statistical significance demonstrating that the switch point of the low-digit participants varied from high-digit participants, though the results were numerically in that direction. We suspect that the number of participants recruited for the main analysis (N=100) is not enough to show the effect we found in experiment 4 (N=178). As this was not the main point of this analysis, we do not pursue it further here, but it is an interesting target for future work.

13 Bounded Fluid Simulation Model (BFS)

13.1 BFS Model Algorithm and Model Fitting Details

Our bounded fluid simulation model is summarized using the pseudocode below:

Algorithm 1 BFS

Require: Number of containers in experiment I, maximum particles $maxParticles$, flow rate $flowRate$, volumes of containers V, duration of stimuli T, particle scaling factor $particleScaling$, intercept $Intercept$, slope $Slope$, fluid simulator Obi

$a \leftarrow Intercept$ ▷ Inherit from the corresponding HLM
$b \leftarrow Slope$ ▷ Inherit from the corresponding HLM
for each container volume $V_i \in [V_1, ..., V_I]$ **do**
  $capacity \leftarrow False$ ▷ Boolean variable indicating reaching $maxParticles$
  $P_{before} \leftarrow 0$ ▷ Number of Standard-sized Particles
  $P_{after} \leftarrow 0$ ▷ Number of Scaled Particles
  $state \leftarrow Obi.initalize()$ ▷ Initialize Fluid Simulator
  $TTF_{simulated} \leftarrow 0$ ▷ The Time Taken for Simulation
  while $TTF_{simulated} \leq T$ **do**
    $TTF_{simulated} \leftarrow TTF_{simulated} + 1$
    if $|state.currentParticles| > maxParticles$ AND $capacity$ is $False$ **then**
      $capacity \leftarrow True$ ▷ Capacity reached
    if $capacity$ is $False$ **then**
      $state = Obi.simulate(size = standard)$ ▷ Forward Simulation
    if $capacity$ is $True$ **then**
      $selected = randomSelect(obi.currentParticles(size = standard))$
      $obi.remove(selected)$ ▷ Remove Selected Particle
      $state = Obi.simulate(size = particleScaling)$ ▷ Forward Simulation
    if $volume(state.currentParticles) \geq V_i$ **then**
      $TTF_{simulated} = (TTF_{simulated} * Slope) + Intercept$
      $P_{before} = Obi.countParticles(size = standard)$
      $P_{after} = Obi.countParticles(size = particleScaling)$
      break
Output: P_{before}, P_{after}, $TTF_{simulated}$

The model can simulate experiments 1, 2, and 4. Experiments 1 and 2 are cases where participants may not reach $maxParticles$, and so do not reach the point of recycling and scaling. In those cases, P_{after} is consistently 0.

For Experiment 2, we assumed the algorithm above is interrupted at some point and returns the current state of the simulator. From this, we need to transform the given amount of particles (an integer amount) into the water-level (measured in pixels on the y-axis). We define the amount of particles that a unit of $WL_{groundTruth}$ represents as WL_{volume}, and this quantity WL_{volume} can be easily calculated as $volume/WL_{groundTruth}$. Notice that this quantity in general depends on the shape of the container, but since we

chose cylindrical containers in which the width and length remained invariant when height increases, we ignore this complication in these studies.

13.2 Bayesian Inference Framework

Given the BFS model above, we formulate a Bayesian inference framework to infer the joint distribution of the two unknown parameters, *maxParticles* and *particleScaling*.

For ease of notation below, we refer to the two parameters as $\theta = \{\theta_1, \theta_2\} = \{maxParticles, particleScaling\}$, and to the experimental data as y. The data y is a $M * N$ matrix, with M participants, and N trials. We refer to the simulated data produced by our BFS model (given parameters θ) as H, a vector with length N.

We are interested in the joint posterior distribution over model parameters, $P(\theta|y,H)$:

$$P(\theta|y,H) \propto P(H|\theta,y)P(\theta|y) \tag{6}$$

We used the Random Walk Metropolis Algorithm (a specific kind of Metropolis algorithm) to infer the joint posterior. We sampled from the joint posterior distribution using the Random Walk Metropolis Algorithm 10000 times. The first 5000 samples are considered as discarded burn-in MCMC samples, and the second 5000 samples are considered as tuned MCMC samples. Only the 5000 tuned MCMC samples were used for analysis.

For the Random Walk Metropolis Algorithm to run, we need to formulate the proposal distribution and the likelihood function.

The proposal distribution: We used the normal distribution as the proposal distribution for each parameter θ_1 and θ_2. The proposal distributions are independent of the data y, and the proposal distributions of θ_1 and θ_2 are independent of one another.

The proposal is updated at each MCMC step. So, at each MCMC steps $t \in [1, ..., 10000]$, we draw:

$$\theta_t \sim MVN(\theta_{t-1}, \Sigma) \tag{7}$$

Where θ_{t-1} is the previous parameter setting proposed at step $t-1$. For initialization at $t = 0$, $\vec{\theta}_0$ is a tunable hyperparameter that determines the initial values of θ.

Σ is constant throughout the whole sampling process, and we define

$$\Sigma = \begin{bmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{bmatrix}$$

The parameters σ_1^2 and σ_2^2 are two tunable hyperparameters that control the random walk drift. The likelihood function: Given the empirical distribution of the experiment data y, we assumed that at each trial $n \in [1, ..., N]$:

$$y_{.,n} \sim N(\mu, s^2) \tag{8}$$

Where μ is the sample mean of observed $y_{.,n}$, and s^2 is the sample variance of observed $y_{.,n}$

So, we assume that for our simulated data H:

$$H \sim MVN(\vec{\mu}, \sigma^2 I_N) \tag{9}$$

Where $\vec{\mu}$ is a vector of length N, which contains the sample means of y, and σ^2 is a tunable hyperparameter, which controls on average how close H has to be with the experiment data y.

In other words, the simulated data from the model is scored in a way that penalizes it the further away it is from the observed data, where the penalty is given by a normal distribution centered on the observed data, with a variance tuned to the data.

The BFS model simulates H given θ. So, the likelihood function is:

$$P(H|\theta, y) = \prod_{n=1}^{N} \phi(H_n; \theta, \vec{\mu}, \sigma^2) \propto \exp\left(-\frac{1}{2\sigma^2}((H-\vec{\mu})^T(H-\vec{\mu}))\right) \tag{10}$$

Where ϕ is the standard normal PDF.

Given the proposal distribution and likelihood function, the Random Walk Metropolis Algorithm infers the desired joint posterior distribution. The pseudocode for the inference is shown below. To avoid numerical error, the likelihood functions are in log scale.

Algorithm 2 Random Walk Metropolis Algorithm

Require: initial θ_0, data y, number of samples T, hyper-parameters Σ, σ^2

for each MCMC sample $t \in [1, ..., T]$ **do**
  $\theta' \sim MVN(\theta_{t-1}, \Sigma)$ ▷ Draw a parameter proposal
  $H_t = BFS(\theta', y)$ ▷ Simulate H_t from BFS with θ'
  Calculate $\log P(H_t|\theta_t, y)$ and $\log P(H_{t-1}|\theta_{t-1}, y)$ ▷ Calculate the Likelihoods
  $r \sim Uniform(0, 1)$ ▷ Draw Acceptance Ratio
  if $\log P(H_t|\theta_t, y) - \log P(H_{t-1}|\theta_{t-1}, y) < \log(r)$ **then** ▷ Accept or Reject
    $\theta_t = \theta'$
  else
    $\theta_t = \theta_{t-1}$
  Output: MCMC Samples

References

Chen, Y.-L., & Lee, Y.-C. (2019). Effects of lighting, liquid color, and drink container type on volume perception. *i-Perception*, *10*(5), 2041669519880916.

Chen, Y.-L., & Shi, L.-J. (2017). Effects of container elongation on consumers' volume perception. *Neuropsychiatry*, *7*(4), 342–346.

Conway, A. R., Kane, M. J., Bunting, M. F., Hambrick, D. Z., Wilhelm, O., & Engle, R. W. (2005). Working memory span tasks: A methodological review and user's guide. *Psychonomic bulletin & review*, *12*, 769–786.

de Paula, J. J., Malloy-Diniz, L. F., & Romano-Silva, M. A. (2016). Reliability of working memory assessment in neurocognitive disorders: a study of the digit span and corsi block-tapping tasks. *Brazilian Journal of Psychiatry*, *38*, 262–263.

Kruschke, J. (2014). Doing bayesian data analysis: A tutorial with r, jags, and stan.

McElreath, R. (2018). *Statistical rethinking: A bayesian course with examples in r and stan*. Chapman and Hall/CRC.

Parsons, S., Kruijt, A.-W., & Fox, E. (2019). Psychological science needs a standard practice of reporting the reliability of cognitive-behavioral measurements. *Advances in Methods and Practices in Psychological Science*, *2*(4), 378–395.

Piaget, J. (2013). *Child's conception of number: Selected works vol 2*. Routledge.

Raghubir, P., & Krishna, A. (1999). Vital dimensions in volume perception: Can the eye fool the stomach? *Journal of Marketing research*, *36*(3), 313–326.

Vehtari, A., Gelman, A., & Gabry, J. (2017). Practical bayesian model evaluation using leave-one-out cross-validation and waic. *Statistics and computing*, *27*, 1413–1432.

Yao, Y., Vehtari, A., Simpson, D., & Gelman, A. (2018). Using stacking to average bayesian predictive distributions (with discussion). *Bayesian Analysis*, *13*(3), 917–1003.