

Working memory control

Modeling task-general knowledge

Michel van de Hoef

1798065

November 2017

Master Thesis

Human-Machine Communication

University of Groningen, The Netherlands

Internal supervisors:

dr. J.P. Borst (Artificial Intelligence & Cognitive Engineering, University of Groningen)

prof. dr. N.A. Taatgen (Artificial Intelligence & Cognitive Engineering, University of Groningen)



university of
 groningen

faculty of mathematics and
 natural sciences

artificial intelligence

Abstract

In today's scientific research, computational models are used to test and expand our theories on human cognition. The models, however, are mostly used to study tasks individually and thus, generalization between computational models is limited. In the present body of work, we examined working memory control, which we theorized to be task-general. Individuals typically rely on one of two control strategies when exercising cognitive control, namely on a proactive or a reactive control strategy, according to the dual-mechanisms of control (DMC) framework. In the current study, three tasks were performed on which participants were expected to benefit from a proactive strategy. In addition, computational models were developed in the PRIMs cognitive architecture, to assess whether variations in performance were the result of differences in the participants' cognitive control strategies. We found that participants who are proactive perform more accurately than those who are reactive on two out of three tasks. On the last task, results were marginally significant. In addition, we demonstrated the proactiveness index to measure more than the mere difference between fast and slow performing participants. The model results, however, did not confirm the variations in performance to be caused by the participants' control strategies. Two explanations were proposed, which are not mutually exclusive. Either individuals mainly adopt proactive control with the strategy to prepare rather than the strategy to rehearse, or the two control modes of the DMC framework did explain only parts of the variations in performance that we found in the behavioral data.

Keywords: cognitive control; working memory control; dual-mechanisms of control framework; task-general knowledge; cognitive modeling; cognitive architecture; PRIMs

Table of Contents

ABSTRACT.....	2
INTRODUCTION	6
PRIMS: INTRODUCTION	8
COGNITIVE CONTROL AND THE DMC FRAMEWORK.....	9
CURRENT STUDY	11
METHODS	14
PARTICIPANTS	14
MATERIALS	14
PROCEDURE.....	14
<i>Names task</i>	15
<i>RITL task</i>	17
<i>CWM task</i>	19
DATA ANALYSIS	22
<i>Outlier analysis: RT data</i>	22
<i>Outlier analysis: Accuracy data</i>	22
RESULTS	23
NAMES TASK	23
RITL TASK.....	25
CWM TASK.....	28
COGNITIVE CONTROL STRATEGY	30
<i>RITL task</i>	31

Working memory control	4
<i>Names task</i>	31
<i>CWM task</i>	32
WORKING MEMORY CAPACITY	33
PROACTIVE VS REACTIVE CONTROL.....	34
MODELING OF RESULTS	35
PRIMs: BASIC PRINCIPLES	35
PRIMs: MODEL DESIGN AND KEY MECHANISMS	37
<i>The RITL task</i>	38
<i>The CWM task</i>	42
PRIMs: MODEL RESULTS	44
<i>The RITL task</i>	44
<i>The Names task</i>	46
<i>The CWM task</i>	48
PRIMs: PROACTIVE VS REACTIVE CONTROL	49
DISCUSSION	51
IMPLICATIONS AND LIMITATIONS	53
FUTURE RESEARCH.....	57
CONCLUSION	59
REFERENCES.....	60
APPENDIX A: QUESTIONNAIRE.....	63
APPENDIX B: STIMULI RITL TASK.....	65
APPENDIX C: INSTRUCTIONS RITL TASK	67

APPENDIX D: STIMULI CWM TASK.....	68
--	-----------

Introduction

Cognition has been studied extensively in the field of psychology. A variety of tasks (e.g. the Tower of Hanoi & the Stroop task) have been used to examine the knowledge, strategies and skills necessary to perform those tasks. Results of the behavioral experiments, however, were limited to the participant's response time and accuracy, but more importantly, the phenomena were largely studied individually, which, according to Newell (1973), would never result in a unified theory of cognition.

To test and expand our theories on human cognition, computational models were developed using so-called “cognitive architectures”, with which we can simulate human cognition. Consequently, assumptions regarding the cognitive processes (e.g. preparing for a stimulus) that produce task behaviors could be tested and evaluated. The term cognitive architecture was first introduced into cognitive science by Bell and Newell (1971). Anderson (2007), in his book “How can the human mind occur in the physical universe?”, defines a cognitive architecture as “a specification of the structure of the brain at a level of abstraction that explains how it achieves the function of the mind.” (p. 7). In this context, the term “function of the mind” relates to human cognition. Thus, to reproduce human behavior, an architecture should encompass the structure of the brain – at the relevant level of abstraction – as well as the cognitive processes. ACT-R (Adaptive Control of Thought - Rational) (Anderson, 2007), for example, maps components of its architecture (i.e. modules) to specific regions of the brain. The declarative module, used in both information storage and retrieval, has been mapped to the prefrontal cortex, for instance. In ACT-R, a total of eight modules (i.e. visual, aural, manual, vocal, imaginal, declarative, goal & procedural) have been linked to the brain (Anderson, 2007).

Cognitive architectures have been mostly used to study tasks individually. For each task, a model is devised with knowledge representations specific to that task. We distinguish between two knowledge types; declarative (factual knowledge) and procedural knowledge (how to perform a task). Two different tasks might rely, partially, on the same knowledge representations (i.e. task-general knowledge). When tasks are modeled separately, however, generalization between the models is limited. Singley and Anderson (1985) found evidence for transfer (of knowledge representations) between two different text editors. When trained on the first type of editor, the participants' performance on the second editor increased significantly. Hence, transfer between tasks does exist and should be reflected in the corresponding cognitive models. But what about tasks that do not share any surface characteristics? Is there transfer to be found between such tasks as well?

In the current study, we have examined working memory control using the primitive elements of information processing (PRIMs) theory (Taatgen, 2013b). We theorized that working memory control is task-general as it is needed in many different situations. We have conducted an experiment in which participants were asked to perform three different tasks. Computational models have been developed in PRIMs, in which we have implemented the dual-mechanisms of control (DMC) framework (Braver, Gray, & Burgess, 2007). We theorized that the DMC framework explains variations in the participants' performance over all three tasks. In the upcoming section, we will first briefly introduce the basics of PRIMs. Then we will cover working memory control and the dual-mechanisms of control (DMC) framework. Next, we will introduce the three tasks that were performed and last, we will cover our hypotheses.

PRIMs: Introduction

PRIMs – a cognitive architecture that was specifically developed to model task-general knowledge – is an extension to the ACT-R architecture (Anderson, 2007), but also incorporates fundamentals of the global workspace model (Dehaene, Erszberg, & Changeux, 1998) and the neural network model by Stocco, Lebiere and Anderson (2010).

Similar to most cognitive architectures, PRIMs is built around production rules, which consist of condition-action pairs. When the conditions of a single production rule are met, the corresponding actions are performed. In the PRIMs cognitive architecture, however, these task-specific production rules are broken down into primitive information processing elements (PRIMs), which perform the most basic operations, such as comparing two items. Hence, the primitive elements are task-general. It is important to note that the PRIMs and their order of execution are stored in an operator. In PRIMs, an operator is the counterpart of a production rule. In the process of learning, the primitive elements are combined through a process called production compilation. The resulting operators remain task-general, as long as these elements do not contain any details specific to a task (Taatgen, 2013a, 2013b). Hence, PRIMs is ideally suited to generalize between multiple models, which is the aim of the current project. An additional advantage of PRIMs over other cognitive architectures is the option to include multiple goals in the model (Taatgen, Katidioti, Borst, & Van Vugt, 2015). PRIMs has been successful in predicting transfer in an arithmetic study (Taatgen, 2013a) and modeling visual distraction (Taatgen et al., 2015).

Cognitive Control and the DMC framework

The domain of task-general knowledge that we have examined is that of cognitive control, which, according to Miyake et al. (2000), consists of three components: the ability to switch between multiple tasks (shifting), the capacity to suppress dominant, automatic responses (inhibition) and the ability to encode new information and keep relevant information active (updating). All three components posit a unique contribution to a participant's performance on a task. The study by Miyake et al. (2000), however, was performed at the level of latent variables. Hence, we know that all three components affect performance differently, but we do not know exactly how. In the present body of work, we examined the dual-mechanisms of control (DMC) framework (Braver et al., 2007) which approaches cognitive control differently; it moves towards the exploration of the intrinsic variability of cognitive control and distinguishes between two control modes. Moreover, the DMC framework examines cognitive control in working memory and with that, aims to explain variations in working memory function. We will proceed with a brief introduction into working memory, which will set the stage for a thorough description of the DMC framework.

A bulk of research exists on working memory, the cognitive system that is used to temporarily store information, and the nature of it. It is established, for example, that working memory is limited in its capacity (Cowan, 2000). Although many postulate working memory to be part of the cognitive architecture (i.e. it is innate), we argue that some aspects are not. Young children, for example, do not make use of rehearsal when trying to remember a sequence of letters (Cowan & Vergauwe, 2014). Rehearsal, the subvocalized repetition of relevant items, is a strategy that is used to improve our working memory capacity (WMC). Without rehearsal, recall

scores decrease. As rehearsal is not innate and is used in multiple tasks, it is a task-general strategy that is learned along the way.

Several theorists (Braver, Gray & Burgess, 2007; Kane & Engle, 2002) indeed posit that variations in WMC are due to differences in strategies regarding the active maintenance of relevant information. According to the DMC framework (Braver, 2012; Braver et al., 2007), individuals tend to develop and rely on two different control strategies when exercising cognitive control, namely a proactive and a reactive control strategy. Whereas an individual who applies a proactive strategy tries to anticipate an upcoming event (i.e. future-oriented & early selection), an individual who applies a reactive strategy waits for an event to happen before deciding on its next action (i.e. past-oriented & late correction). Thus, proactive control requires internal preparation and reactive control is solely triggered by external events.

Furthermore, proactive control requires reliable, predictive contextual cues to be present in the environment. If contextual cues are misleading, preparations could end up to be futile and then, proactive control would be too costly. In addition, a strong goal-relevant focus is maintained during proactive control, and goal-relevant information is held active. In contrast, during reactive control an individual engages in increased goal-irrelevant processing (e.g. the individual is distracted by the environment). Hence, the proactive control strategy is more resource demanding and thus, reactive control is easier to apply.

As proactive control is more resource demanding, Braver et al. (2007) hypothesize that individuals who possess a greater working memory span are more prone to engage in proactive control, but only if they might benefit from it. In short, they stipulate that the construct to measure WMC is based on the same underlying mechanism as the construct to measure cognitive

resources. Consequently, individuals with a greater WMC would have more resources available and thus, should be able to engage in proactive control more regularly.

Although situational factors (e.g. the expected working memory load) have been shown affect an individual's preferred cognitive control strategy (Braver, 2012), in the present body of work we assumed that the participant's control mode was set and did not change over the course of the experiment. In other words, we expected participants to either adopt a proactive or a reactive control strategy across all three tasks that were performed throughout the experiment.

Current Study

In this thesis, we have answered the following research question: In what way does the participant's cognitive control strategy predict the performance across different problem-solving tasks? Three tasks were performed: The Names task (Anderson, Qin, Jung, & Carter, 2007), the RITL task (Cole, Bagic, Kass, & Schneider, 2010) and a complex working memory (CWM) span task (Chein & Morrison, 2010). Subjects were to benefit from a proactive control strategy on all three tasks, as in all tasks goal-relevant information had to be held active to successfully complete the task: In the Names task, participants had to memorize three names and an instruction; in the RITL task, subjects had to maintain and combine different instructions; and in the CWM task, participants had to memorize various letters.

All three tasks will be explained in detail in the methods section. The RITL task, however, will be introduced briefly here since we hypothesized that the participant's mean response time (RT) on the RITL task is an indication of the participant's cognitive control strategy (i.e. proactive or reactive), and we used it as a "proactiveness index".

In the RITL task, participants were provided with three rule components that had to be combined. To give an example, the combination *same*, *green* and *left index* meant: If the answer to the question “*green?*” is the *same* for both stimuli, press your *left index* finger. Thus, participants had to memorize the rule components and consecutively, apply those to the stimuli presented on the screen. Each instruction set was used in three consecutive trials, after which a new combination of rule components was introduced. In between the rule components and the first stimuli, a white screen appeared for 2.0s. We hypothesized that proactive participants would use this timeframe of 2.0s to start combining the three instructions, whereas reactive participants would simply wait until the presentation of the first stimuli before starting to combine the instructions. Given that proactively combining the instructions should lead to faster responses, participants who replied fast after the onset of the first stimuli scored higher on the proactiveness index and thus, were considered proactive. In contrast, participants who were slower to reply after the onset of the first stimuli were lower on the proactiveness index and thus, were considered reactive. See Figure 1 for a graphical illustration of the proactiveness index.

To summarize, we expected the participants’ cognitive control strategies to be task-general and therefore, participants with a proactive control strategy to perform more accurately on all three tasks than participants who use a reactive control strategy, as in all three tasks a

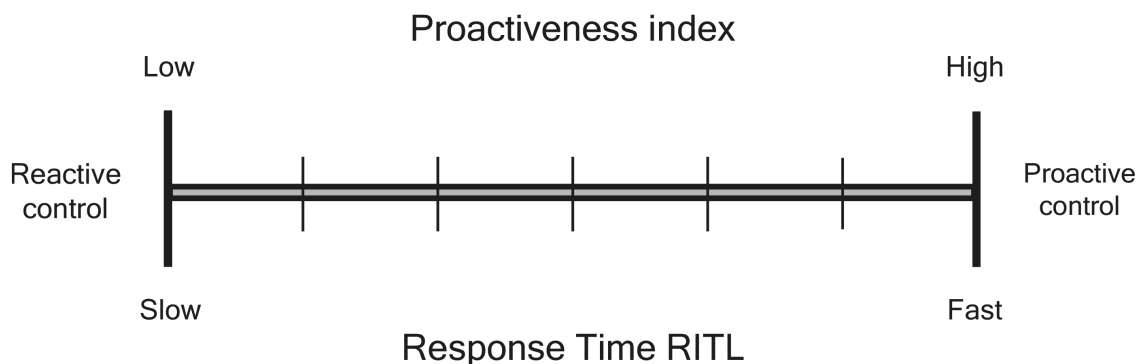


Figure 1: Visualization of the proactiveness index, for which we used the participants’ mean response times on the first trials of the RITL task.

proactive strategy is beneficial. In addition, we expected participants high in WMC to be more prone to engage in proactive control than participants low in WMC. With that, we formulated the following hypotheses: firstly, participants high on the proactiveness index, who we consider proactive, perform more accurately on the RITL task as well as the Names task and the CWM task than participants low on the proactiveness index, who we consider reactive; secondly, our implementation of the two control modes in the computational models explains the variations in performance found in the behavioral data of all three tasks; and lastly, participants with a high accuracy score on the CWM task, who thus are high in working memory capacity, score higher on the proactiveness index than participants with a low accuracy score on the CWM task, who are low in working memory capacity (Braver, 2012; Braver et al., 2007).

The first two hypotheses have been examined using the behavioral data from the experiment together with the computational models developed in PRIMs. Proactive participants (i.e. low mean RT on the first trials of the RITL task) were expected to show better accuracy scores on all three tasks than reactive participants (i.e. a high mean RT on the first trials of the RITL task). In addition, the model results were to fit the behavioral data. The computational models of both the Names task and the CWM task used the strategy to rehearse to achieve proactive control, whereas the model of the RITL task used the strategy to prepare. When adopting reactive control, all models just waited for the next event to happen. The last hypothesis has been tested using the behavioral data only. Participants with a high accuracy score on the CWM task were expected to score higher on the proactiveness index than participants with a low accuracy score on the CWM task.

Methods

Participants

The study included students at the University of Groningen and the Hanze Hogeschool between the ages of 18 and 30. Participants were recruited via the Facebook group “Paid research participants Groningen” and were offered €16,- for participation. All participants filled in an informed consent. In total, fifty-seven subjects participated in the study of which fifty were found eligible for data analysis (see “Outlier analysis: Accuracy data” on p. 22 for details; 16 male; age 22.9 ± 2.9).

Materials

Relevant keyboard keys were labeled with colored stickers. Also, a questionnaire was administered. It was constructed to assess the participants’ demographics and control strategies used. The questionnaire is reported in Appendix A.

Procedure

Three tasks were performed; the Names task (Anderson et al., 2007), the RITL task (Cole et al., 2010) and a complex working memory (CWM) span task (Chein & Morrison, 2010). The tasks were presented in counterbalanced order – creating 6 groups – to control for order effects. The experiment took 2 hours per participant. In the upcoming section, the experimental set-up will be described in detail.

Names task

Task

The Names task (Anderson et al., 2007) was developed in OpenSesame v3.1.4 (Mathôt, Schreij, & Theeuwes, 2012). For this task, participants were instructed to memorize 8 two-letter word / two-digit number pairs (e.g. IN – 12). The word-number associations were used as instructions which indicated in what way participants had to change the order of three names (e.g., Dick, Fred, Tom) that were presented on the screen. To give an example, the instruction 31 meant that the participants had to switch around the third and first name (i.e. Tom, Fred, Dick). Similarly, 23 indicated that the participants had to rearrange the second and third name (i.e. Dick, Tom, Fred). The instruction could also consist of a two-letter word (e.g. IN) that corresponded to a two-digit number. Then, participants first had to retrieve the word-number combination before rearranging the names. Moreover, the instruction could be inapplicable (i.e. a “no-op”). For example, the instruction 41 could not be applied, as there is no fourth name. In that case, the participant just had to repeat the names in their original order. See Figure 2 for an overview of the Names task.

The names task consisted of 4 conditions, namely no transformation / no substitution (e.g. 14), transformation / no substitution (e.g. 23), no transformation / substitution (e.g. BY) and transformation / substitution (e.g. AN). All conditions contained 4 unique instructions. In total, the experiment consisted of 4 blocks of 16 trials. Hence, each condition got presented 16 times.

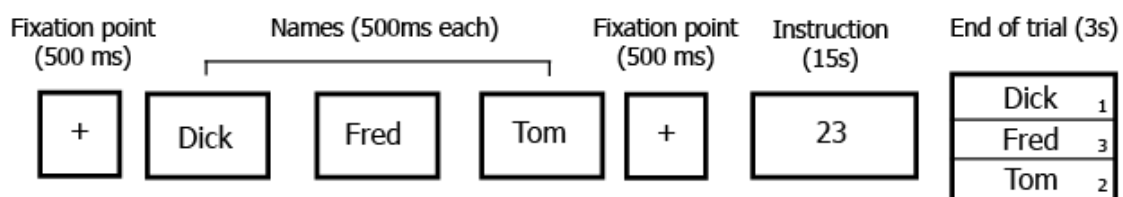


Figure 2: Schematic depiction of a single trial in the Names task.

The dependent variables were RT, measured from the onset of the instruction to the moment the subject indicated to be ready for answer, and the participant's accuracy.

Training

Prior to the test, participants were to memorize the 8 word-number associations. After 10 minutes of self-paced paper-and-pencil training, subjects were shown a word on the computer and were asked to respond by typing the corresponding number. Each block consisted of 8 word-number combinations and lasted until all 8 were answered correctly, i.e. mistakes were repeated. The training session consisted of 4 blocks.

Practice

At the onset of the Names task, participants had to practice the procedure. In total, 6 practice trials were performed. In the first three trials the instructions were number-based, and in the remaining three trials the instructions were word-based.

Experimental set-up

A fixation point indicated the onset of a trial (500ms), after which the names were presented - in random order - for 500ms each. Then, another fixation point (500ms) appeared and the instruction was shown. Once participants knew their answer, they had to click the left mouse button to proceed. When after 15.0s still no response was registered, the task continued as well. Then, a display appeared, depicting all three names in random order. The subjects could provide their answer by selecting all three names in their correct order. After 3.0s, the display disappeared as well as the time window for a subject to reply.

Feedback

Participants received feedback after each trial. If the answer was correct, the feedback text was presented in green. If the answer was either incorrect or the participant responded too slow, the feedback text was depicted in red.

RITL task

Task

The RITL task was adopted from Cole et al. (2010), and further developed using E-Prime v2.0. In the RITL task, participants were instructed to perform small tasks. Each task consisted of three rule components that were combined in different ways, creating unique tasks. In total, 12 rule components were permuted to develop 64 tasks. Out of the 64 tasks, 4 tasks were presented repeatedly, both during practice and the test. A task consisted of a logic rule (i.e. same, just one, second & not second), a semantic rule (i.e. sweet, loud, soft & green) and a motor response rule (i.e. right middle, right index, left middle & left index). To give an example, the combination same, sweet and right index meant: If the answer to the question “sweet?” is the same for both of two stimuli, press your right index finger. Participants had to memorize the rule components and consecutively, apply those in three trials. In each trial, two stimuli were presented on the screen, to which the rule components had to apply. See Figure 3 for an overview of a single task. The dependent variables in the RITL task were RT, measured from the onset of the stimuli to the subject’s response, and the participant’s accuracy.

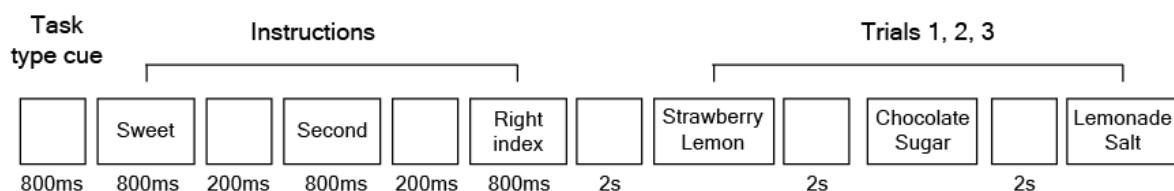


Figure 3: Schematic depiction of the RITL task.

Stimuli

The stimuli that we used in the RITL task (i.e. 45 per semantic category; 180 in total) were adopted from Cole et al. (2010) and analyzed by 4 reviewers prior to the test. Several stimuli were found to be too difficult for non-native speakers to understand. Therefore, those stimuli were replaced with ones more commonly known. The stimuli and their replacements are depicted in Appendix B.

Instructions

A pilot study indicated the RITL task to be complex and hard to understand. The logic rules, in particular, were not fully understood and even after several revisions, the instruction sheets did not suffice. Therefore, we decided to develop a walkthrough in addition to the instruction sheets that all participants had to undertake, before proceeding with the practice sessions. The walkthrough explained all logic rules, response rules and their implications step-by-step, in which no stage could be skipped. Results did indicate the participants' understanding of the logic rules and response rules to have improved considerably after implementation. The final version of the instruction sheets is reported in Appendix C.

Practice

Prior to the test, participants had to perform a practice session, in which only the repeated tasks were used. The practice session consisted of 30 tasks in total, divided over 5 blocks. In the first 4 blocks of practice, accuracy feedback was provided directly after each trial. In the last block, feedback was provided only at the end.

Test

At the onset of the test session the distinction between novel (thin borders) and repeated tasks (bold borders) was explained. Once participants acknowledged that they did understand these

instructions, they proceeded with the test. In total, the test consisted of 28 repeated tasks and 28 novel tasks, presented in 7 blocks (8 tasks per block). Hence, each participant had to perform 28 out of the 60 unique tasks. The novel tasks were selected on a semi-random basis, in that all 4 logic rules were presented equally often. Accuracy feedback was provided at the end of each block.

Experimental set-up

Both during practice and the test, a task started with the task type cue (i.e. thin or bold border), presented for 800ms, indicating whether it was going to be a repeated or a novel task. Then, the three rule components were introduced, all presented for 800ms. The rule components were presented in a pre-defined order; logic rule, semantic rule and response rule. In between the presentation of the three rule components as well as the rule components and the task type cue a white screen was depicted for 200ms. Following the task type cue, the screen remained blank for 2.0s before the two stimuli of the first trial were depicted. Once participants knew their answer, they could respond by pressing 2 (left middle finger), 3 (left index finger), 8 (right index finger) or 9 (right middle finger). Once the response was registered, 2.0s passed before the stimuli of the second and consecutively, the third trials were shown. In between tasks, 2.0s passed before the instructions of the upcoming task were presented.

CWM task

Task

The CWM task (Chein & Morrison, 2010) was adopted from Daamen (2016), and further developed using PsychoPy v1.82.01. In the CWM task, participants were instructed to memorize a sequence of letters (i.e. the storage task) while performing a lexical decision task (i.e. the choice task). Participants were instructed to perform both tasks equally well. In the storage task,

letters had to be memorized one at a time. The letter span ranged from 3 to 5. Only consonants were used and these were randomly selected. In the choice task, subjects were asked to provide an answer to the question “Does this object fit in a shoebox?” to a variety of items (e.g. building & picture). The items, which are depicted in Appendix D, were adopted from Daamen (2016), and translated into English. In total, the list of stimuli consisted of 50 words that were to be answered positively and 50 words that were to be answered negatively.

The test consisted of 4 blocks of 9 trials. Hence, each letter span was presented 12 times. The dependent variables were the subjects’ accuracy on the choice task and the partial-credit unit score (PCS) (Conway et al., 2015) on the storage task. The PCS is calculated, per trial, as the percentage of letters that is answered in the correct serial position (i.e. items correct / span * 100)¹. See Figure 4 for an overview of the CWM task.

Practice phase

At the onset of the CWM task, participants had to perform a practice session. Subjects performed 6 practice trials, before proceeding with the test. The practice trials were of letter span 3 only. The choice task items that were used during practice, which were different from the ones used in the test, were adopted from Daamen (2016) and translated into English.

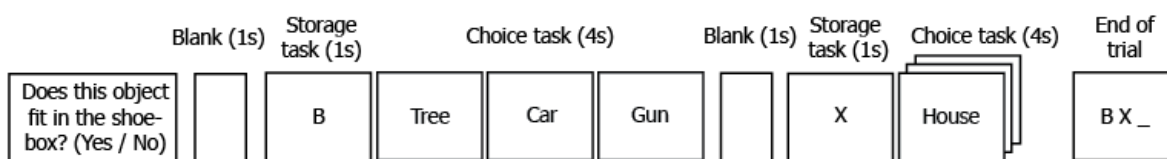


Figure 4: Schematic overview of a single trial in the CWM task. In the storage task, one letter is presented at a time. In the choice task, items are shown until 3.9s have passed.

¹ Scoring with a partial-credit unit method also tends to create a normal distribution in the data (Conway et al., 2015).

Storage and processing phase

At the onset of each trial the question “Does this object fit in a shoebox?” was depicted on the screen. Participants were to press the spacebar to proceed. Then, a white screen was depicted for 1.0s after which the first letter was presented (1.0s). After that, the first item of the choice task was shown to which participants had to reply with the left ctrl button (no) or the right ctrl button (yes). New items of the choice task were presented until 3.9s had passed. Notably, the last item of the choice task was cut-off at the 3.9s mark. Participants did not have to respond to that item anymore. Then, after 100ms the next letter was presented on the screen (1.0s). This process was repeated until all letters of the letter span were shown.

Recall phase

In the recall phase, participants had to fill in the letters in their correct serial position. The total number of letters to respond with was indicated with underscores. The underscores were replaced with the letters the subjects replied with. Participants could adjust their answer using the backspace button. Once the letter span was accounted for, the task proceeded with the next trial.

Feedback phase

After each block participants received feedback on their accuracy scores in both the choice task and the storage task. In addition, feedback was provided on the participants’ RTs in the choice task.

Data analysis

Data analysis was performed in R (R Core Team, 2015). We performed an outlier analysis on the RT data for both the Names task and the RITL task and an outlier analysis on the accuracy data for all three tasks.

Outlier analysis: RT data

Incorrect responses were excluded from RT data analysis. Response times below 300 milliseconds were discarded from the analysis in both the Names task and the RITL task.

Furthermore, in the RITL task responses above 15000 milliseconds were removed from the analysis. Lastly, in both tasks, responses that were more than 3 standard deviations from the mean, calculated per condition, were excluded from the analysis.

Outlier analysis: Accuracy data

In the Names task, participants were excluded from the analysis when they scored below 60% on no transformation / no substitution condition, as a subject simply would have to reply with the current names in their original order. In the RITL task, participants with an accuracy score below 50% were discarded from the analysis as this would indicate a subject to not perform the task. In the CWM task, participants were excluded from the analysis if they scored below chance level (50%) on the choice task or when their accuracy score was below 60% on the storage task. In total, 7 participants were excluded from the analysis.

Results

In the present study, we aimed to investigate the effects of the participants' cognitive control strategies on their performance of three tasks². Firstly, we will examine the results of all three tasks separately. Then, we will consider the cognitive control strategies that were used. Next, we will analyze the effects of the subjects' working memory capacity on the cognitive control strategies they adopted and lastly, we will briefly study the differences in RT on the RITL task between proactive and reactive participants.

Names task

Figure 5 depicts the mean RT in milliseconds per condition on the Names task. The mean RT was the lowest in the No Retrieval / No Transformation condition and the highest in the Retrieval / Transformation condition. As the RT data was nonlinear, data analysis was performed using a

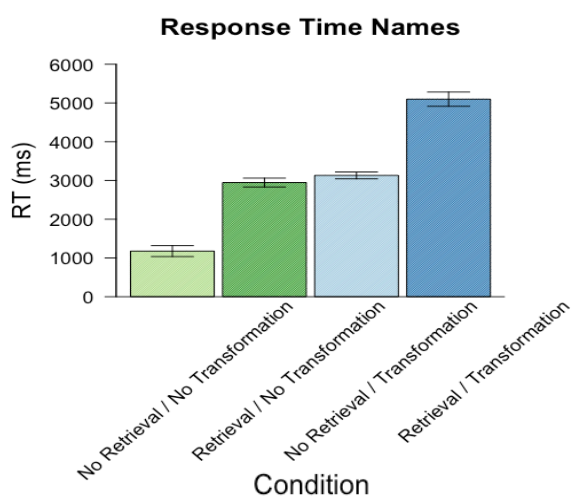


Figure 5: Mean response time (RT) in milliseconds per condition on the Names task depicted with the within-subjects SE.

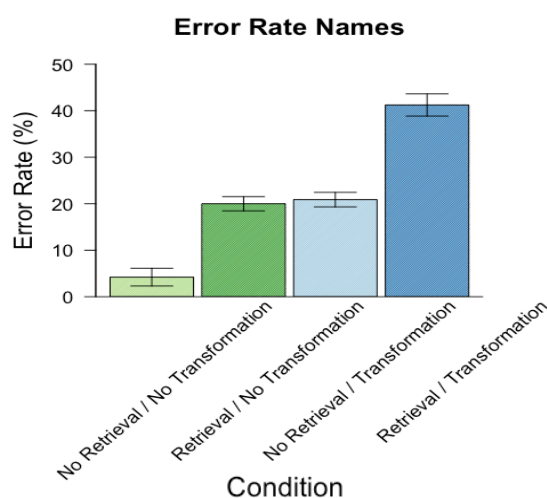


Figure 6: Mean error rate per condition on the Names task depicted with the within-subjects SE.

² Preliminary data analysis showed no significant effect of task order on the performance indicators of all three tasks.

log-transformation. A mixed-effects linear regression analysis with subject as a random factor confirmed a main effect for both Retrieval ($\beta: 0.84; t(2436) = 34.89; p < 0.001$) and Transformation ($\beta: 0.88; t(2436) = 36.58; p < 0.001$)³. Thus, when participants either had to perform a retrieval request or reorder the names, the mean RT increased significantly.

Figure 6 shows the mean error rate per condition on the Names task. The mean accuracy was lowest in the Retrieval / Transformation condition and the highest in the No Retrieval / No

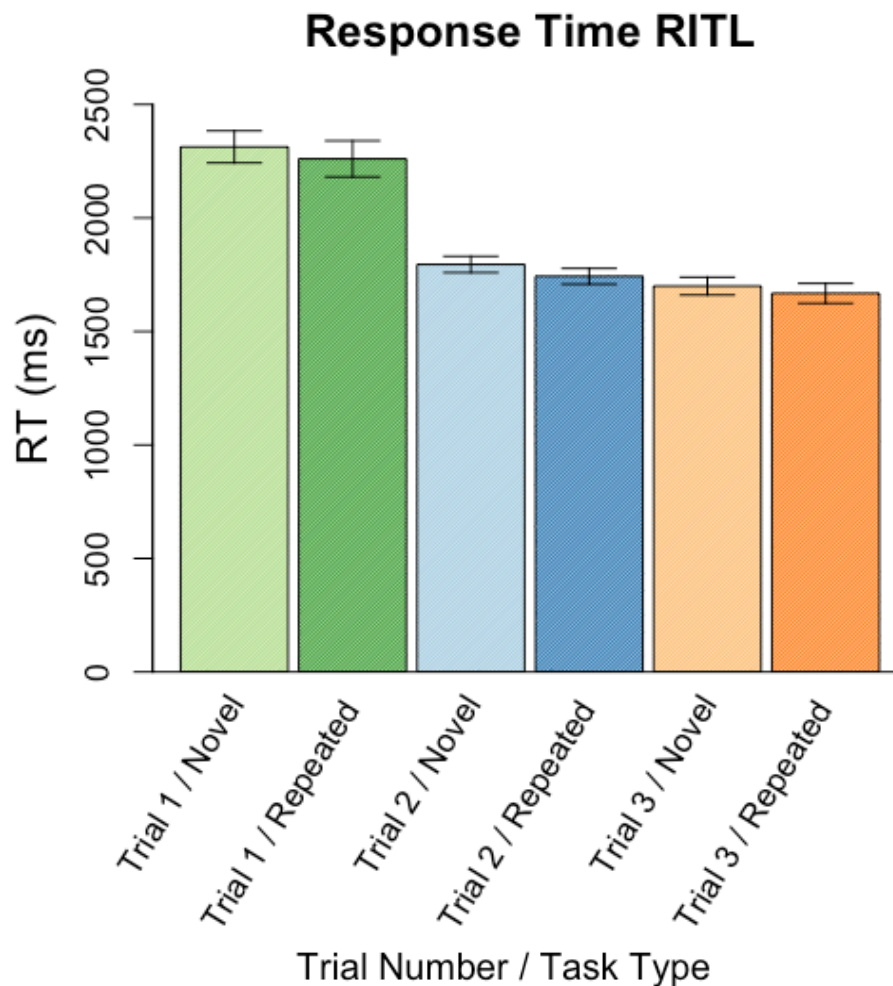


Figure 7: Mean RT in milliseconds per trial number (1, 2 & 3) and task type (novel & repeated) on the RITL task depicted with the within-subjects SE.

³ In the mixed-effects linear regression analyses we used Satterthwaite's approximations to the degrees of freedom (Satterthwaite, 1946), which are based on the number of subjects in the analysis rather than the number of groups.

Transformation condition. A mixed-effects logistic regression analysis showed a main effect for both Retrieval (β : -1.86; $z = -9.15$; $p < 0.001$) and Transformation (β : -1.92; $z = -9.48$; $p < 0.001$). When participants either had to retrieve a word-number combination or transform the names, the mean accuracy decreased significantly. These findings were as expected, as they replicated the results from the original study with the Names task (Anderson et al., 2007).

RITL task

The mean RT per task type, depicted per trial number, is shown in Figure 7. Overall, participants were slower to respond in the first trials of a task compared to the second and third trials.

Moreover, the within-subjects SE was evidently the largest in the first trials, which posits a first indicator that our assumption to take only the first trials for the proactiveness index⁴, is correct.

To test the relationship between Task type and RT we performed a mixed-effects linear regression analysis over all 3 trials with a log-transformation on the RT data, as the data was found to be nonlinear. Results showed the difference between novel and repeated tasks to be significant ($\beta = -0.03$; $t(6468) = -2.54$; $p = 0.01$). Participants replied faster when performing repeated tasks ($M = 1888$; $SE = 23$ ms) than novel tasks ($M = 1931$; $SE = 23$ ms).

The mean error rate per trial type is depicted in Figure 8. The participants' error rate did not vary greatly over the 3 trials, nor did the within-subjects SE. Results from a mixed-effects logistic regression analysis over all 3 trials did indicate the difference between novel and repeated tasks to be significant (β : 0.11; $z = 1.98$; $p = 0.048$). Participants performed better on the

⁴ In the first trial of an instruction set (i.e. task) of the RITL task, participants had to actively combine the rule components and consecutively, apply the instructions to the two stimuli presented on the screen. In the second and third trials, participants merely had to re-apply the same instructions. Therefore, we theorized that the participants' RT on the first trial of an instructions set would be the best indicator for our proactiveness index.

RITL task on repeated tasks ($M = 0.80$; $SE = 0.01$) than on novel tasks ($M = 0.78$; $SE = 0.01$).

These findings replicated the results as reported in Cole et al. (2010).

Next, we examined the relationship between Logic cue and the performance indicators of the RITL task. These analyses did solely include the responses on the first trials, as those responses were used to determine the proactiveness index. Figure 9 shows the mean RT per logic cue. Participants responded fastest on tasks with the logic cue “second”, and the slowest on tasks containing the logic cues “same” and “not second”. The within-subjects SE was largest when participants had to perform tasks containing the logic cues “same” and “not second”. We

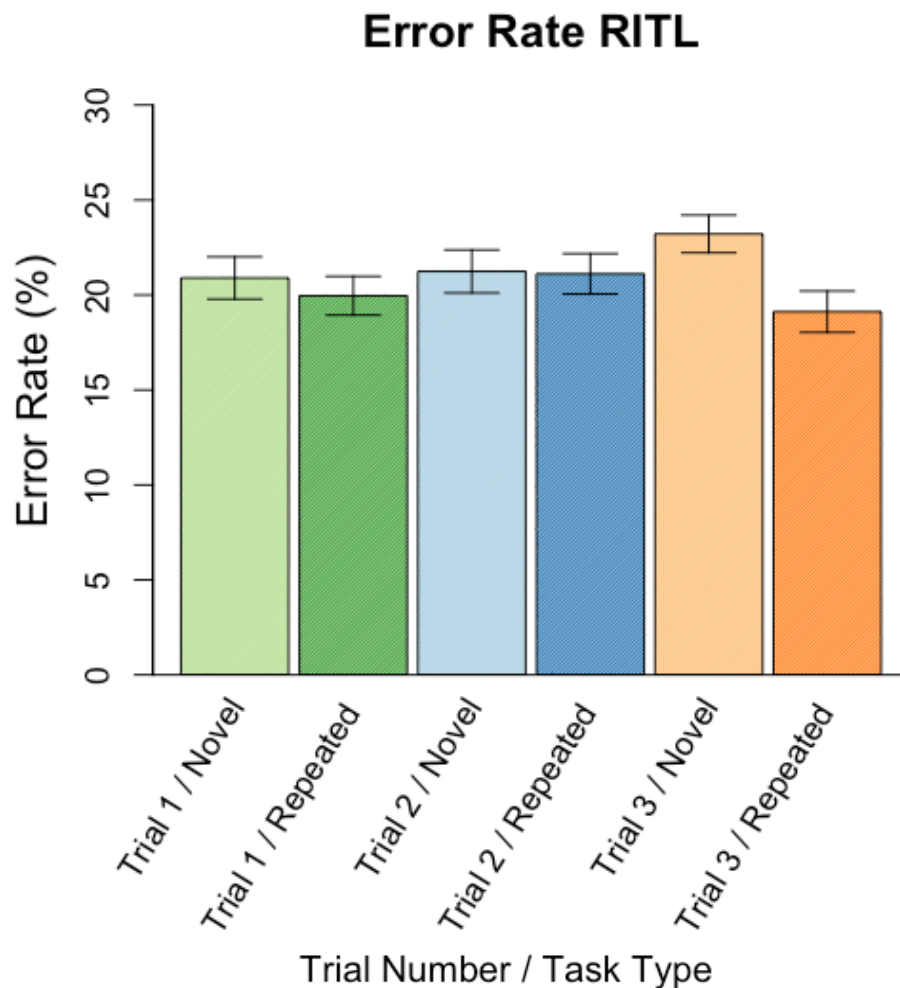


Figure 8: Mean error rate per trial number (1, 2 & 3) and task type (novel & repeated) on the RITL task depicted with the within-subjects SE.

performed a mixed-effects linear regression analysis containing the first trials with a log-transformation on the RT data, and with the logic cue “same” as the reference group. Results confirmed a significant difference with the logic cues “second” ($\beta: -0.37; t(2096) = -13.09; p < 0.001$) and “just one” ($\beta: -0.11; t(2096) = -3.7; p < 0.001$) and a marginally significant difference with the logic cue “not second” ($\beta: -0.05; t(2096) = -1.8; p = 0.07$). Thus, when compared to the tasks containing the logic cue “same”, participants responded faster on tasks with the cues “second” and “just one” and we found inconclusive evidence for the difference with tasks containing the logic cue “not second”.

The mean error rate per logic cue is depicted in Figure 10. The participants’ accuracy scores were highest on tasks containing the logic cue “second” and the lowest on tasks with the cues “same” and “not second”. The outcomes of a mixed-effects logistic regression analysis with the logic cue “same” as reference group confirmed a significant difference with all other logic cues: “second” ($\beta: 1.29; z = 8.60; p < 0.001$), “just one” ($\beta: 0.82; z = 6.03; p < 0.001$) and “not second” ($\beta: 0.39; z = 3.08; p = 0.002$). Hence, whereas tasks containing the cues “same” and “not

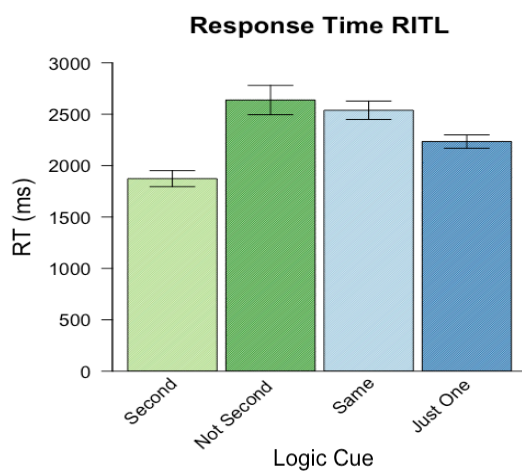


Figure 9: Mean RT in milliseconds per logic cue on the RITL task depicted with the within-subjects SE.

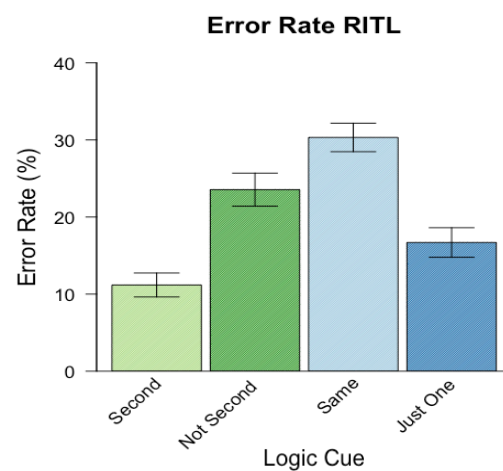


Figure 10: Mean error rate per logic cue on the RITL task depicted with the within-subjects SE.

second” were the most difficult to perform, tasks with the cue “second” were easier. The level of difficulty of tasks with the cue “just one” was in between.

CWM task

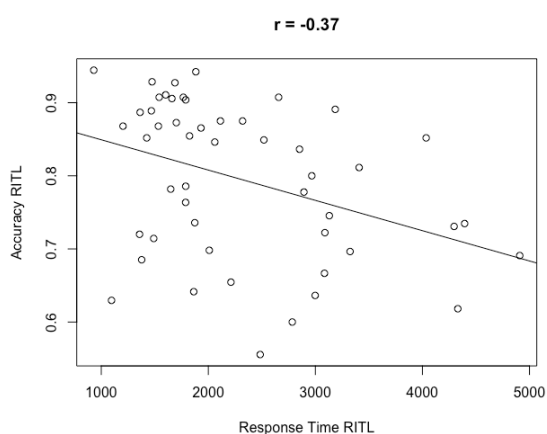
For the CWM task, we calculated the mean partial-credit unit score (PCS) (Conway et al., 2015), i.e. the number of correct items divided by the letter span, on the storage task, which is depicted in Figure 11. The subjects’ mean PCS was the highest on letter span 3 and the lowest on letter span 5. A mixed-effects logistic regression analysis showed the relationship between Span and PCS to be significant (β : -0.42; $z = -3.85$; $p < 0.001$). Thus, participants performed worse when the letter span increased, as was expected from Chein and Morrison (2010).



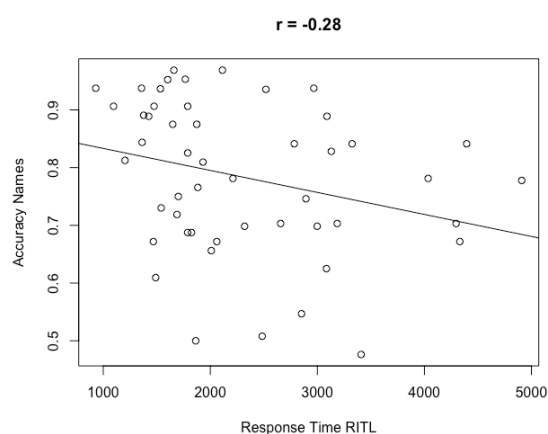
Figure 11: The mean partial-credit unit score (PCS) per span (3, 4 & 5) on the storage task of the CWM task depicted with the within-subjects SE.

Subjects also performed the choice task, of which the results were of minor relevance.

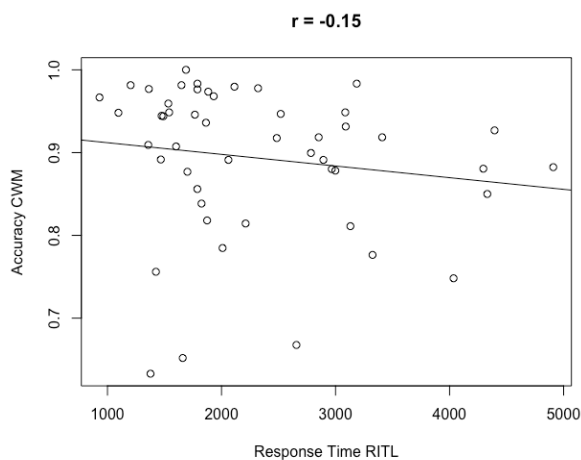
Nevertheless, participants were expected to put effort in the task. The mean accuracy on the choice task was 0.90, ranging from 0.74 to 0.96. Hence, participants were equally devoted to the choice task as the storage task.



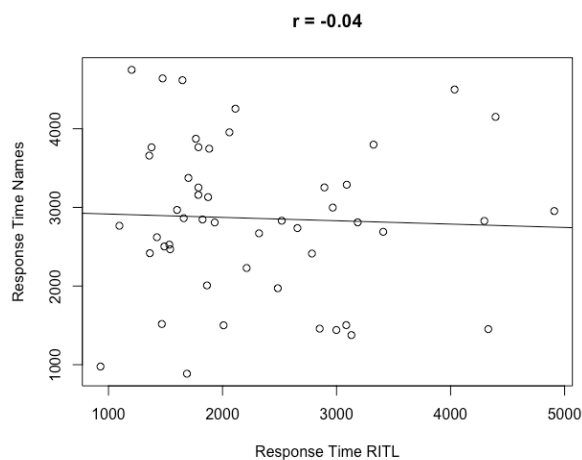
A) Scatter plot of the linear relationship between the proactiveness index and the accuracy on the RITL task.



B) Scatter plot of the linear relationship between the proactiveness index and the accuracy on the Names task.



C) Scatter plot of the linear relationship between the proactiveness index and the accuracy (PCS) on the CWM task.



D) Scatter plot of the linear relationship between the proactiveness index and the RT on the Names task.

Figure 12: Scatter plots depicting the linear relationship between the proactiveness index and the accuracy on all three tasks (A, B & C) as well as the RT on the Names task (D).

Cognitive control strategy

The main aim of the present thesis was to examine the effects of the subjects' cognitive control strategies on their performance of three different tasks. We expected the proactiveness index (i.e. the mean RT on the first trials of the RITL task) to be an indicator of the participants' cognitive control strategies, as fast response times on the RITL task meant for a participant to have prepared for the upcoming stimuli. With that, we hypothesized participants high on the proactiveness index, who we consider proactive, to perform more accurately on the RITL task as well as the Names task and the CWM task than participants low on the proactiveness index, who we consider reactive.

First, we analyzed the relevant scatter plots and correlations, which are depicted in Figure 12. The correlations indicated there to be a negative linear relation between the RT on the first trials of the RITL task and the accuracy / PCS on all three tasks, which corresponds with our hypothesis. The correlation was significant between the RT on the RITL task and the accuracy on the RITL task ($r(48) = -0.37, p = 0.007$), as well as for the linear relationship between the RT on the RITL task and the accuracy on the Names task ($r(48) = -0.28, p = 0.049$) and no evidence was found for a linear relationship between the RT on the RITL task and the accuracy (PCS) on the CWM task ($r(48) = -0.15, p = 0.29$). In other words, participants who were faster on the first trials of the RITL task performed better on the RITL task and the Names task. However, we found no indication of a relationship between the RT on the first trials of the RITL task and the performance on the CWM task.

Then, we investigated the linear relationship between the RT on the RITL task and the RT on the Names task, as we theorized those two to be unrelated⁵. As was expected, the correlation appeared to be absent ($r = -0.04$; $p = 0.77$), see Figure 12. To further analyze the impact of the cognitive control strategies that were adopted by our participants, we added the proactiveness index to the regression analyses of all three tasks.

RITL task

First, we investigated the effects of the proactiveness index on the performance of the RITL task. The analysis was performed on trial level. A mixed-effects logistic regression analysis of the relationship between condition and accuracy on the RITL task was performed that contained Logic cue and the proactiveness index as fixed factors and the intercept per participant as random effect. The effects of the proactiveness index were found to be significant (β : -0.28 ; $z = -2.95$; $p = 0.003$), in that slowly executed trials were performed significantly worse than trials of the RITL task that were executed quickly. When we compared the complete model with the model without the proactiveness index, the reduction in residual sum of squares was statistically significant ($\chi^2(1) = 8.13$; $p = 0.004$). In other words, the model containing the proactiveness index was significantly better than the model without this fixed effect.

Names task

Next, we added the proactiveness index to the mixed-effects logistic regression analysis of the Names task. The resulting model contained the Transformation condition, the Retrieval condition and the proactiveness index as fixed factors and the intercept per participant as random factor.

⁵ Whereas in the RITL task proactive participants could prepare for the stimuli by proactively combining the instructions, in the Names task proactive participants could solely rehearse the instructions. Thus, in the former case proactive participants should perform the task faster than participants who are reactive, but in the latter case the participants' cognitive control strategies should not affect their response times.

The effects of the proactiveness index on the accuracy of the Names task turned out to be significant (β : -0.31; z = -2.21; p = 0.03). Comparing the two models, the reduction in residual sum of squares was significant in the complete model (χ^2 (1) = 4.68; p < 0.03). As predicted, participants who adopted a proactive control strategy performed better on the Names task than participants who used a reactive control strategy.

We also aimed to investigate the effects of the proactiveness index on the RT of the Names task. Therefore, we added the proactiveness index to the mixed-effects linear regression analysis. Results did show the effects of the participants' cognitive control strategies on the RT of the Names task to be nonexistent (β : 0.02; t (2436) = 0.36). Subsequent comparison of both models supported these findings (χ^2 (1) = 0.14; p = 0.71). Thus, the participants' cognitive control strategies did not affect their response times on the Names task⁶.

CWM task

Lastly, we studied the effects of the participants' cognitive control strategies in the CWM task. The mixed-effects logistic regression analysis of the relationship between the letter span and PCS now contained Span and the proactiveness index as fixed factors and the intercept per participant as random factor. The effects of the cognitive control strategies participants adhered to were found to be marginally significant in the CWM task (β : -0.25; z = -1.64; p = 0.10). When we compared the complete model with the model without the proactiveness index, the reduction in residual sum of squares was marginally significant as well (χ^2 (1) = 2.70; p = 0.10). Hence, there is an indication that participants who used a proactive cognitive control strategy performed better on the CWM task than participants who adopted a reactive control strategy, but we did not reach significance.

⁶ As was expected, based on the absence of a linear relationship between the RTs on both tasks.

Working memory capacity

We hypothesized that participants with a high accuracy score on the CWM task, who thus are high in working memory capacity, score higher on the proactiveness index than participants with a low accuracy score on the CWM task, who are low in working memory capacity. In other words, participants with a high accuracy score on the CWM task should show a faster reaction time on the RITL task (i.e. adopt a proactive control strategy) than participants with a low accuracy score on the CWM task. We performed a mixed-effects linear regression analysis of the relationship between Trial type and the log-transformation of the RT on the RITL task, in which Trial type and the mean PCS score per participant on the CWM task were fixed factors and the intercept per participant was a random factor. The results, however, did show the effect of the mean PCS score on the CWM task to be statistically insignificant (β : -0.65; $t(48)$: -1.11; p = 0.27), and so did the comparison between the complete model and the model without the mean PCS score ($\chi^2(1) = 1.27$; $p = 0.26$). Hence, participants with a high working memory span did not seem to automatically engage in proactive control.

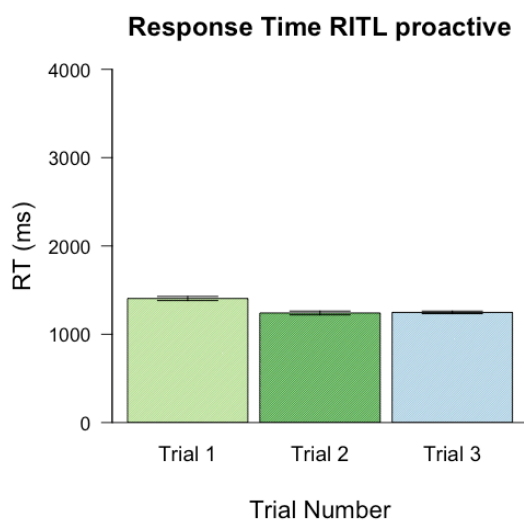


Figure 13: Mean RT in milliseconds per trial number (1,2,3) of proactive participants on the RITL task depicted with the within-subject SE.

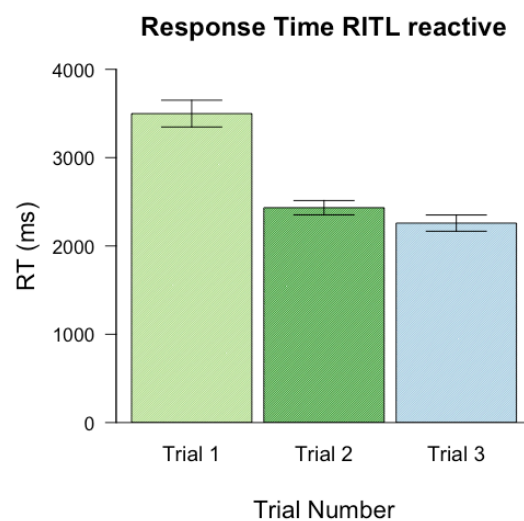


Figure 14: Mean RT in milliseconds per trial number (1,2,3) of reactive participants on the RITL task depicted with the within-subject SE.

Proactive vs reactive control

As we aimed to compare the outcomes of the behavioral data with the model results, we decided to dichotomize the behavioral data into two groups. For both groups, we selected the 15 outermost cases. Hence, our proactive control group consisted of the 15 participants with the lowest mean RT on the first trials of the RITL task and our reactive control group contained the 15 participants with the highest mean RT on the first trials of the RITL task. Figure 13 and 14 show the mean RTs per trial number of proactive and reactive participants, respectively. The mean RTs differed substantially between proactive and reactive participants on the first trials, whereas the distinction was less present in the second and third trials of the RITL task. Hence, these results reconfirmed the first trials of the RITL task to be the best indicator for our proactiveness index. It is worth noting that proactive participants still replied nearly twice as fast as reactive participants in the second and third trials of the RITL task.

Modeling of results

The computational models were developed in PRIMs (Taatgen, 2013b). In the upcoming section, we will start with an introduction of the basic principles underlying PRIMs, after which we will introduce all models and their key mechanisms. Then, we will discuss the model results as well as the fit with the behavioral data.

PRIMs: Basic principles

In the introduction, we already discussed the primitive elements of information processing, the theory on which PRIMs (Taatgen, 2013b) is primarily built, and production compilation – both that make PRIMs ideally suited to generalize between multiple models. Next, we will examine

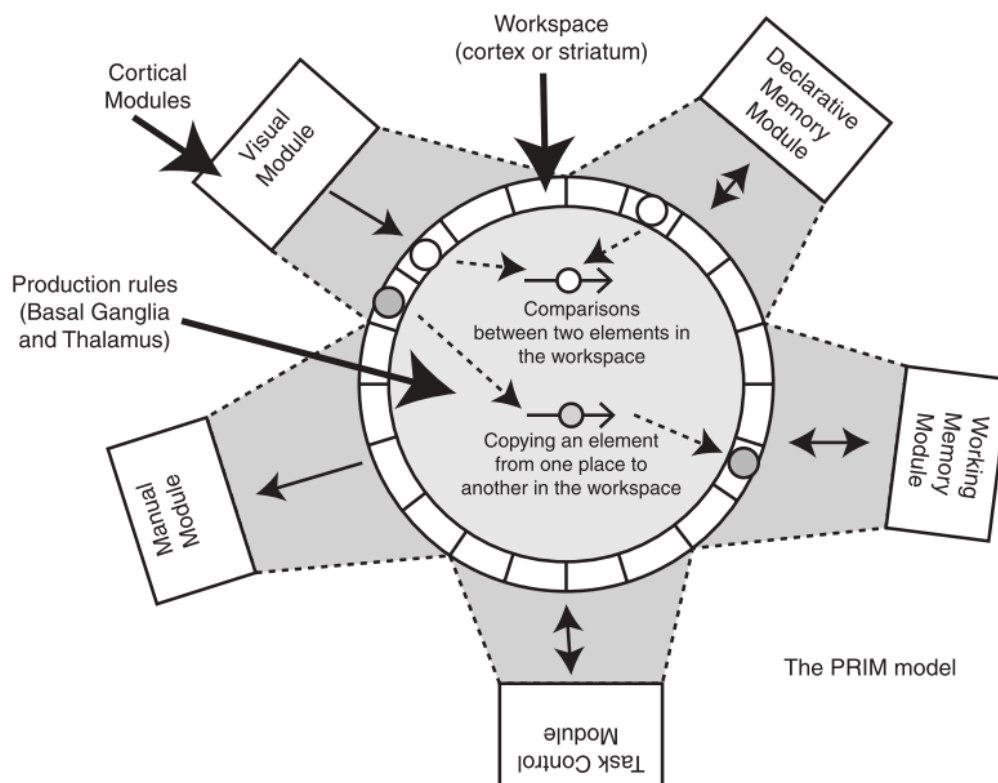


Figure 15: An overview of the primitive elements of information processing (PRIM) theory, as depicted in Taatgen (2013).

the major components of the PRIMs cognitive architecture and discuss the way in which relevant operators are selected.

PRIMs contains five cortical modules – the manual module, the visual module, the declarative module, and the working memory module – and a central workspace, as is shown in Figure 15. A buffer is linked to each module, through which the modules communicate. All buffers together form the central workspace. In the central workspace, PRIMs are used to compare information (i.e. condition PRIMs) and copy information between buffers (action PRIMs). If the condition PRIMs of an operator are met, its corresponding action PRIMs are performed. Figure 16 shows an example of an operator. When the conditions of multiple operators are satisfied, the operator with the highest activation is selected. Operators receive extra activation when they are performed successfully, but also receive spreading information from other sources within PRIMs, such as the input and the goal buffer. The amount of spreading activation that an operator receives from the different sources is determined through a set of parameters (Taatgen, 2013b).

```
operator rehearse {  
  // "When the screen is blank, retrieve the first name"  
  V1 = second-screen-item  
  RT1 = nil  
  G5 = first-goal  
  ==>  
  G0 -> RT2  
  one -> RT3  
}
```

Figure 16: The operator that initiates rehearsal in our models of the Names task. The lines of code above the arrow (==>) represent condition PRIMs and the lines below the arrow are action PRIMs.

PRIMs: Model design and key mechanisms

In this section, each of the models will be explained in detail. We will start with a brief introduction of the way new information is stored and the way we have implemented this in our computational models. Then, we will introduce the parameters that we have adjusted to produce a better model fit and last, we will introduce the models.

Although controversy remains on the way we store sequential information, there is evidence that we are capable of grouping items (chunking) into one unit, while accounting for ordinal information (Dehaene, Meyniel, Wacongne, Wang & Pallier, 2015). We have used this theory and implemented it in each of our computational models with position-specific storage operators, which are operators that store the sequential items, one at a time, that are deemed relevant in a task. The sequential items that are relevant in the RITL task are, for example, the different rule components. In the computational models of the RITL task, each of the rule components is stored in a different knowledge chunk together with its serial position and a reference to the current goal chunk. The reference to the current goal chunk is used to enable the model to retrieve the items that are relevant in a specific trial.

In addition to the position-specific storage operators, we also created position-specific rehearsal and position-specific retrieval operators. In case of a retrieval failure in position-specific rehearsal operators (i.e. the model fails to retrieve one of the sequential items), the model starts over and retrieves the first item again. In case of a retrieval failure in position-specific retrieval operators, the model continues to “guess” the answer. Our implementation does not account for educated guesses – “guess” answers are always deemed incorrect.

In all our models, we adjusted different parameters to produce a better fit with the behavioral data. We varied the retrieval threshold (rt): which determines the minimum activation

at which a chunk still gets retrieved, the latency factor (lf): which regulates the time that is required for a retrieval, and the activation noise parameter (ans): which affects the noise during retrieval. For each task specifically, additional parameters had to be adjusted. Those will be discussed together with the models of that task.

The RITL task

The model of the RITL task needed to be able to combine the rule components and apply those to the stimuli presented on the screen. To do so, several steps had to be performed: read and store the instructions, combine the instructions, apply the instructions to the stimuli and provide a response. Furthermore, we expected participants who adopt a proactive control strategy to

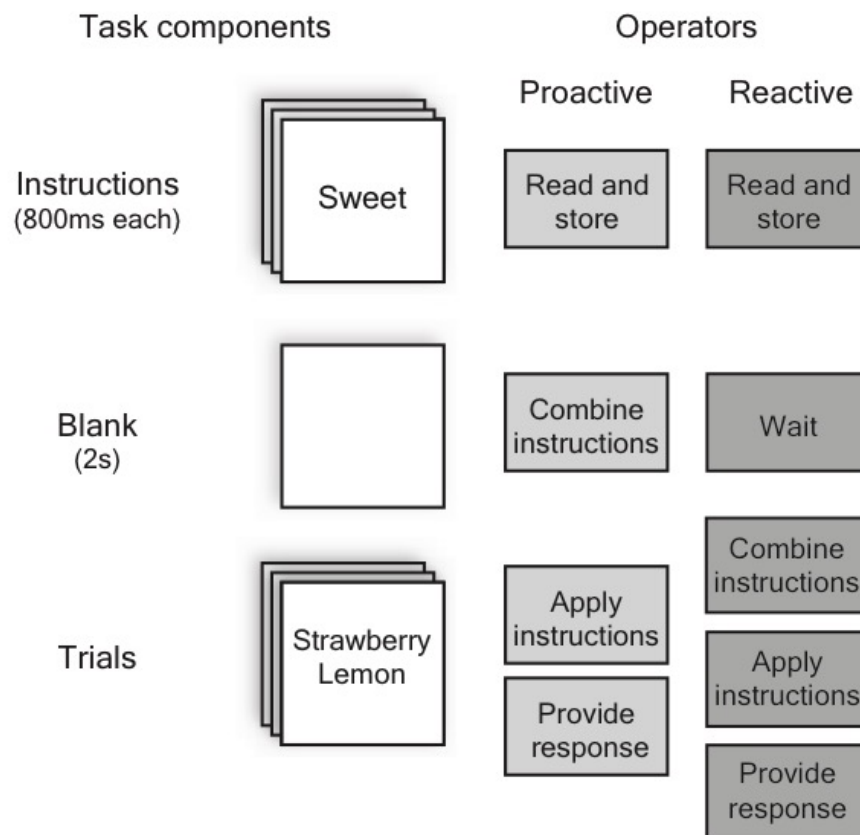


Figure 17: An overview of the computational models of the RITL task. Operators are shown for both the proactive and reactive models.

perform better on the RITL task than participants who use a reactive control strategy. Therefore, we had to distinguish between a proactive and a reactive model of the RITL task. We theorized that the proactive model would use the 2.0s timeframe between the instructions and the first stimuli to start combining the instructions (i.e. the strategy to prepare), whereas the reactive model would simply wait until the presentation of the first stimuli before starting to process the instructions. Figure 17 depicts an overview of the models of the RITL task. The complete models are available upon request.

The model of the RITL task contains position-specific storage and position-specific retrieval operators. In “combine the instructions”, intermediate results are kept in different working memory slots. In case of a retrieval failure, both in “combine the instructions” and “apply the instructions to the stimuli”, the model responds with “guess”. In other words, when an element of the instructions is forgotten, the model guesses the answer. When the model has provided its answer and thus, proceeds to the next trial, the model starts over at “apply the instructions to the stimuli”.

As our participants already had substantial experience with storing and retrieving instructions, we decided to train the models – 48 trials in total – before running the actual experiment. Also, to provide a better fit with the behavioral data, we had to lower the production-prim-latency variable, which, simply put, affects the time needed to fire an operator. In the end, we were unable to model the distinction between novel and repeated trials. As it turned out, we needed to create a hierarchical representation of working memory, which the newest version of PRIMs does support. However, we developed our models in a previous version, which did not include this feature.

The Names task

To perform the Names task, the model needed to be able to memorize all three names together with the instruction. Also, the model had to retrieve the relevant word-number associations from memory as well as to rearrange the names. The following steps were to be performed: read and store the names, retrieve and store the word-number combination (optional), rearrange the names (optional) and provide a response. One more step was required and constituted the difference between a proactive and a reactive control strategy; rehearsal. When the fixation point appeared, the proactive model started to rehearse the three names, whereas the reactive model just waited for the instruction to appear. A schematic overview of the models of the Names task is shown in Figure 18. The complete models are available upon request.

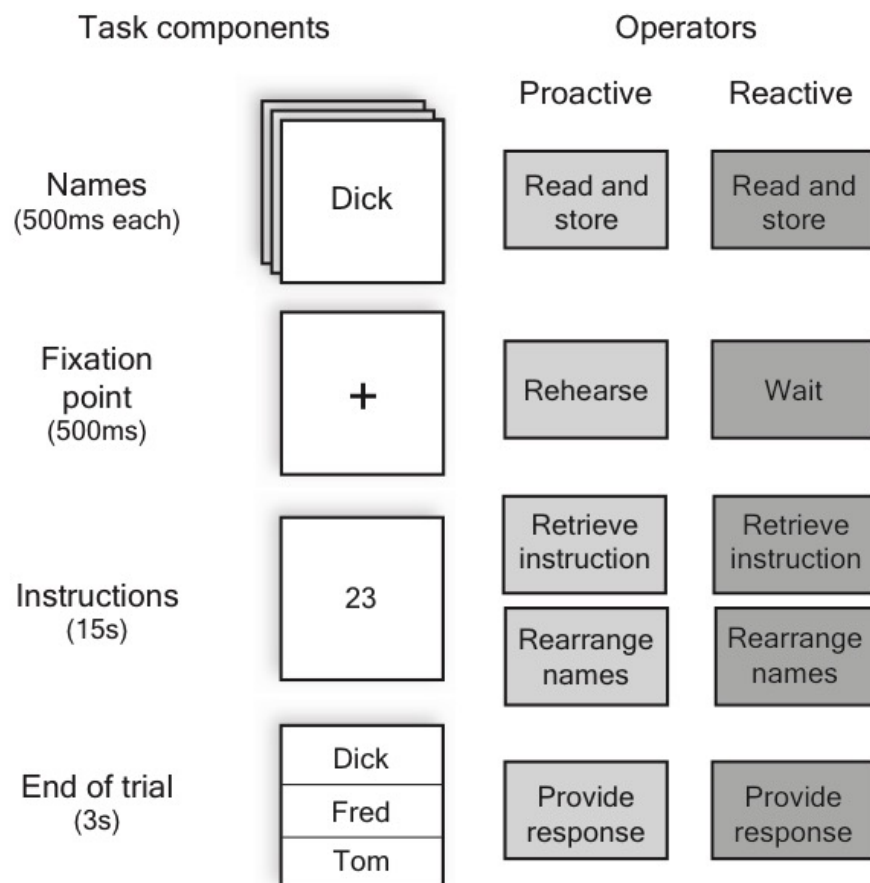


Figure 18: Schematic depiction of the computational models of the Names task. Operators are depicted for both the proactive and reactive models.

The model of the Names task contains position-specific storage, retrieval, and position-specific rehearsal operators. Once the three names are stored in declarative memory, the proactive model starts to rehearse. Then, if the instruction that is shown on the screen is word-based, the model makes a retrieval request for its number-based counterpart and the number-based instruction is kept in a working memory slot. When the number-based instruction contains a four, the model retrieves the names and prepares to answer. However, if the number-based instruction does not contain a four, the model retrieves the names that need to be re-ordered one by one. Then, for each name a new knowledge chunk is stored in declarative memory with its correct, new, serial position together with a reference to the current goal chunk. Once that process is completed, the model retrieves all the names, stores them in working memory slots and prepares to answer. When a retrieval error occurs in “retrieve and store the word-number combination” or “rearrange the names”, the model prepares to answer with “guess”.

Again, as our participants had ample experience with the storage and the retrieval of an instruction, we decided to train the proactive and reactive models – 32 trials in total – before we ran the experiment. Furthermore, we had to adjust the goal-activation parameter, which is used to regulate the spreading activation from the goal.

It turned out to be difficult to produce a good model fit of the Names task. Eventually, as we will see in the “PRIMs: Model results” section, we were able to produce a nice fit of the data taken altogether, but we were unable to successfully implement the distinction between a proactive and reactive control strategy. Many variations have been suggested and tested: store all three names in one knowledge chunk, rather than in separate chunks; keep the names, when retrieved, in their correct, new order in working memory, instead of creating new chunks in declarative memory; first retrieve all three names, before processing the instructions; lower the

latency factor (lf), to make the model spend more time rehearsing; and so on. This issue will be analyzed in detail in the “Discussion” section.

The CWM task

The model of the CWM task needed to perform two tasks simultaneously; the choice task and the storage task. For the choice task, two steps were required: process the word and provide a response. For the storage task, again two steps were needed: read and store the letters and give a response. Equal to the models of the Names task, the proactive control strategy was modeled as rehearsal. An overview of the models of the CWM task is given in Figure 19. The complete models are available upon request.

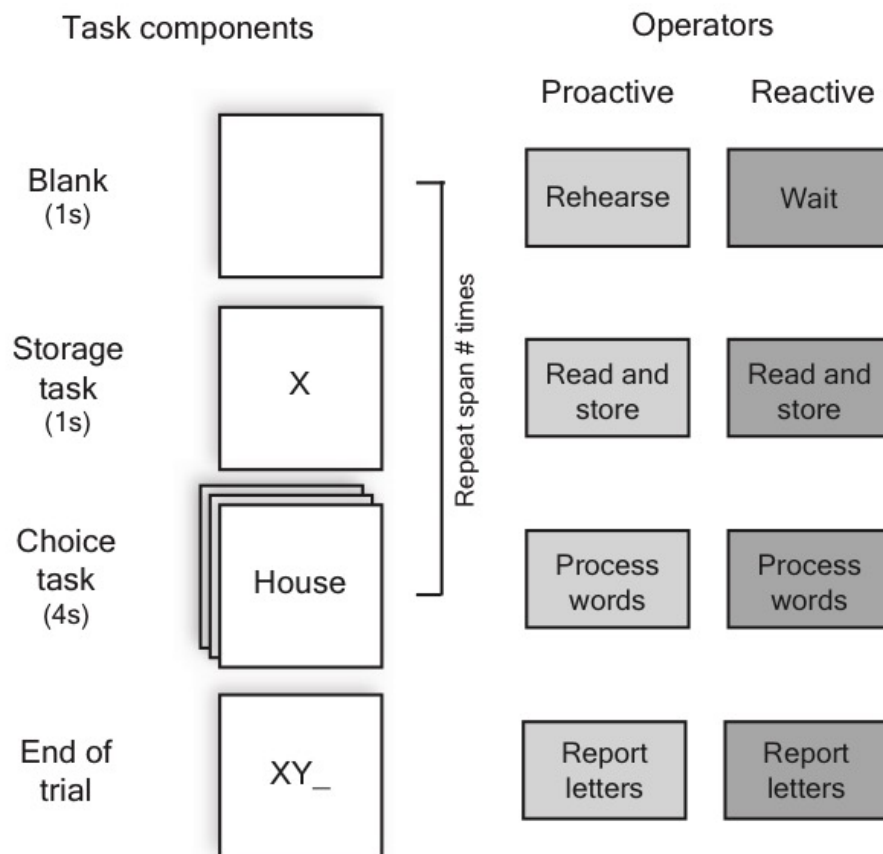


Figure 19: Schematic overview of the computational models of the CWM task. Operators are depicted for both the proactive and reactive models.

The model of the CWM task contains all three types of position-specific operators (i.e. storage, retrieval & rehearsal). The model starts with storing the first letter of the storage task in declarative memory. Then, the model proceeds with the choice task. For each item, a retrieval request is made for its trait (Does it fit in a shoebox?), and the model provides its response. Instead of answering with “guess”, when a retrieval error occurs, the model of the CWM task provides the answer “error”. Once the choice task is finished, the proactive model starts to rehearse. This process is repeated for the complete letter span. At the end of a trial, the model simultaneously retrieves the letters in their correct serial position and provides its answer.

We decided to train the models for 3 trials before running the experiment. Furthermore, in the behavioral experiments, the choice task was presented for 3.9s and after that, a white screen appeared for 100ms before the next letter was shown. We decided to extend the choice task with 100ms in our computational models. As a result, no blank screen had to be presented in between the choice task and the next letter of the storage task⁷. We also had to adjust one of the parameters, the input-activation, which affects the spreading activation from the input buffer.

⁷ In the current version of PRIMs, the script waits until an operator is completely run, before proceeding with the next task component. Thus, when the model retrieves a rehearsal operator while the screen is blank for 100ms, that complete operator is run. As a result, the model might miss the first few hundred milliseconds while the next letter is presented. We decided to overcome this problem by leaving out the blank screen.

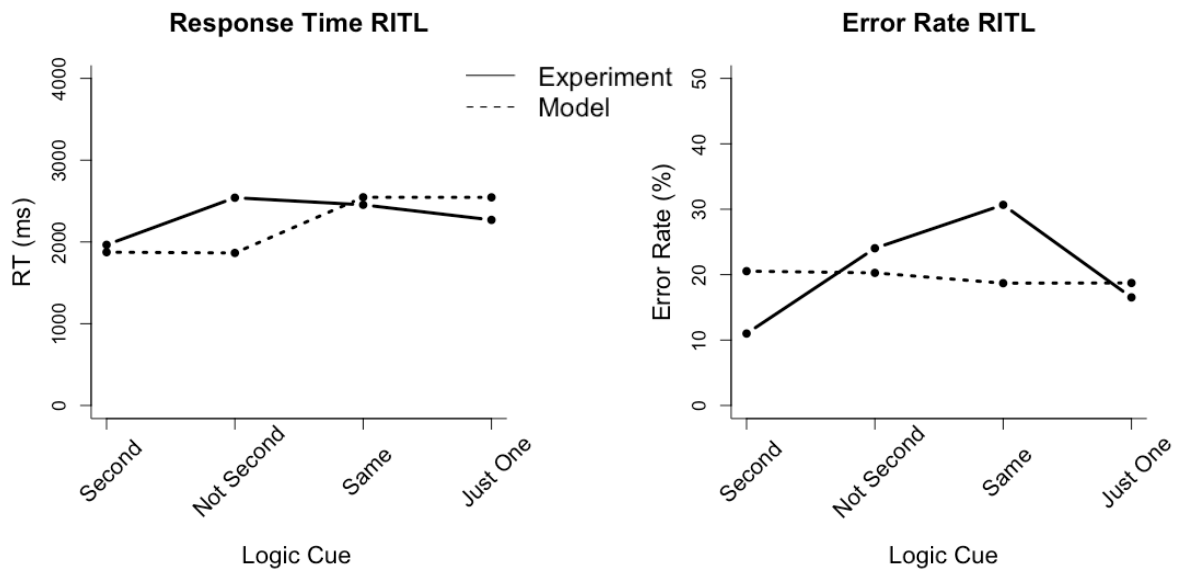
PRIMs: Model results

We ran the models 500 times and compared the model outcomes to the results of the behavioral experiments. We only compared the model outcomes to those results from our behavioral data that were found to be significant. However, as the effects of the proactiveness index on the PCS of the CWM task were marginally significant, we decided to analyze those results as well. We will start with the model results of the RITL task, after which we will proceed with the model outcomes of the Names task and last, we will analyze the model results of CWM task. We will finalize this section with a brief discussion concerning the effects of both control modes in our model data.

The RITL task

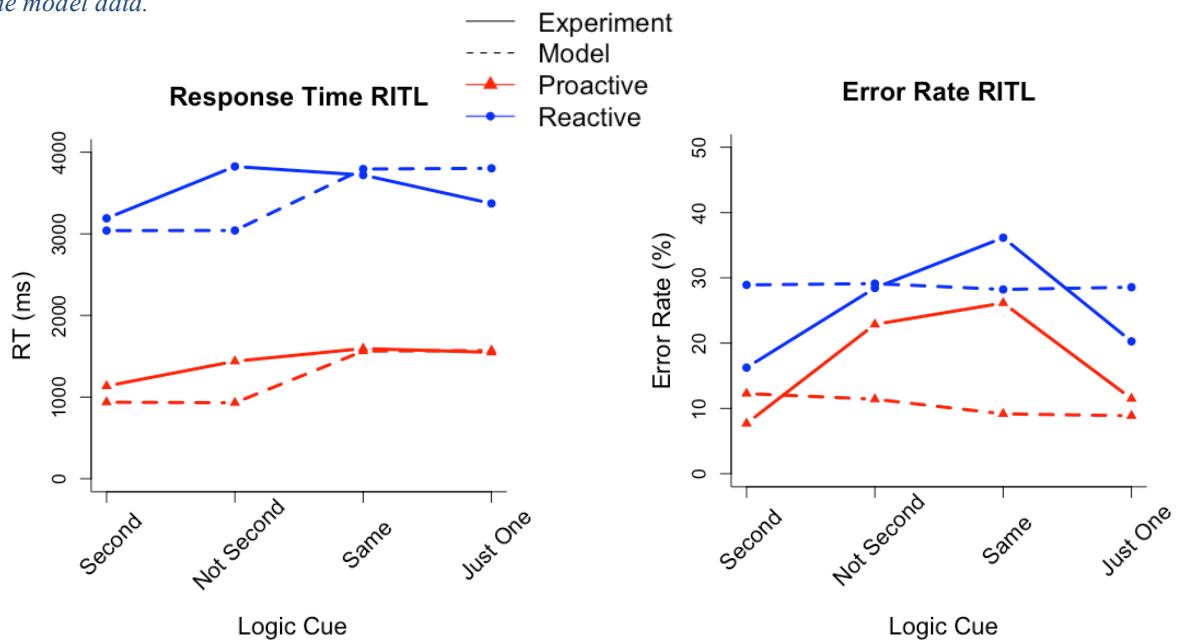
The outcomes of the models that were developed for the RITL task are shown in Figure 20, together with the outcomes of the behavioral experiments. Firstly, we examined the RT (20A) and error rate (20B) of all participants. The model data fitted the behavioral data quite nicely, regarding both the RT and the error rate. We were unable, however, to model the differences between the logic rules that we found in the behavioral data.

Next, we examined the model results per cognitive control strategy. In Figure 20C (RT) and Figure 20D (error rate), the model data is plotted against the behavioral data of proactive and reactive participants. The proactive and reactive models mirrored the distinction between a proactive and a reactive control strategy that we found in the experimental data nicely; the proactive model both scored better on the RITL task and performed the task faster than the reactive model. Again, we were unable to model the differences that we found in the behavioral data between the logic rules.



A) Mean RT in milliseconds on the RITL task depicted for all participants. The solid line represents the behavioral data and the dotted line the model data.

B) Mean error rate on the RITL task depicted for all participants. The solid line shows the behavioral data and the dotted line the model data.



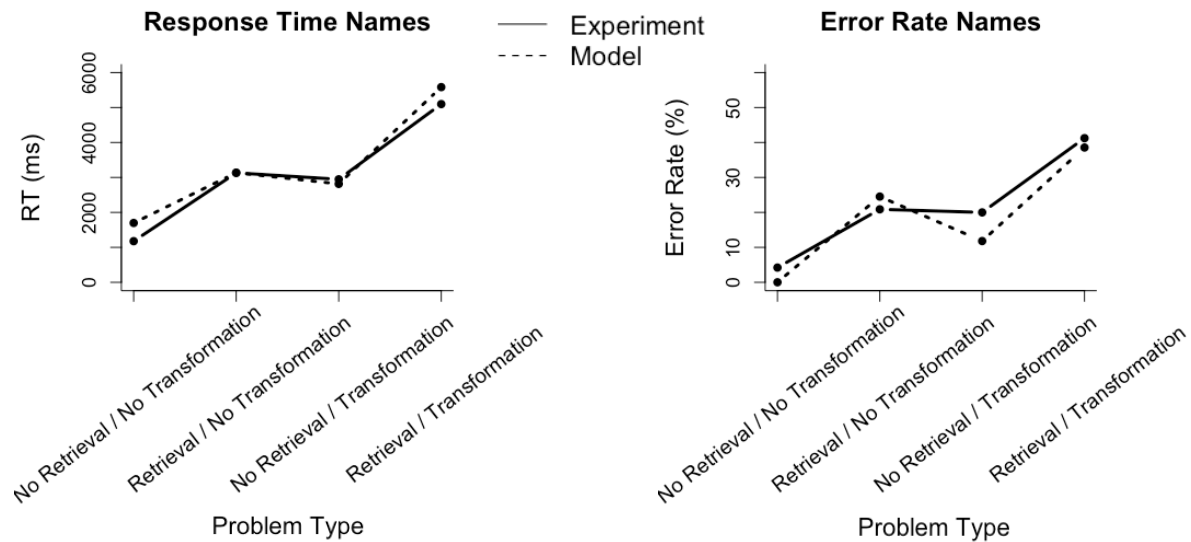
C) Mean RT in milliseconds on the RITL task depicted per control strategy. Proactive control is shown in red and reactive control in blue. The solid line represents the behavioral data and the dotted line the model data.

D) Mean error rate on the RITL task depicted per control strategy. Proactive control is shown in red and reactive control in blue. The solid line represents the behavioral data and the dotted line the model data.

Figure 20: Comparison of the outcomes between the behavioral data and the model data. Data from all participants is used in plots A & B. The data is dichotomized in proactive and reactive control in plots C & D.

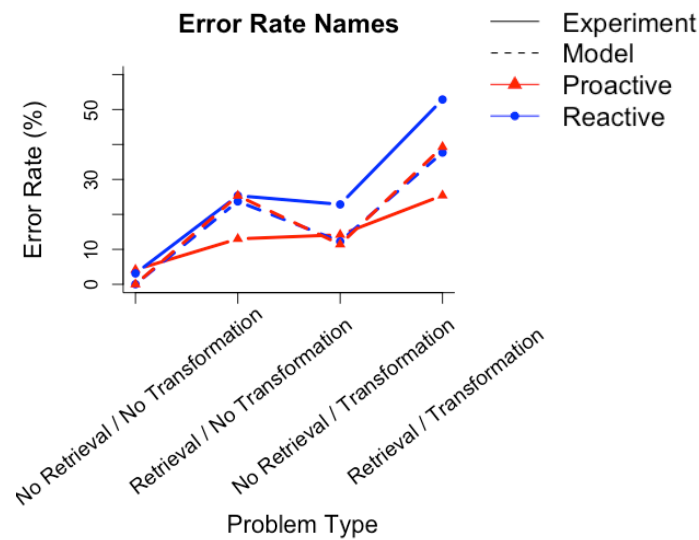
The Names task

The model outcomes of the Names task are depicted in Figure 21, together with the results of the behavioral experiments. The mean RT per condition in the model data did mirror the mean RTs that we found in the behavioral data of all participants (Figure 21A), as did the model's mean error rate (Figure 21B). When we compared the error rates from the proactive and the reactive models to those of proactive and reactive participants, however, no similarities were found. Even after several revisions, we were unable to model any distinction between a proactive and reactive participant.



A) Mean RT in milliseconds on the Names task depicted for all participants. The solid line represents the behavioral data and the dotted line the model data.

B) Mean error rate on the Names task depicted for all participants. The solid line represents the behavioral data and the dotted line the model data.

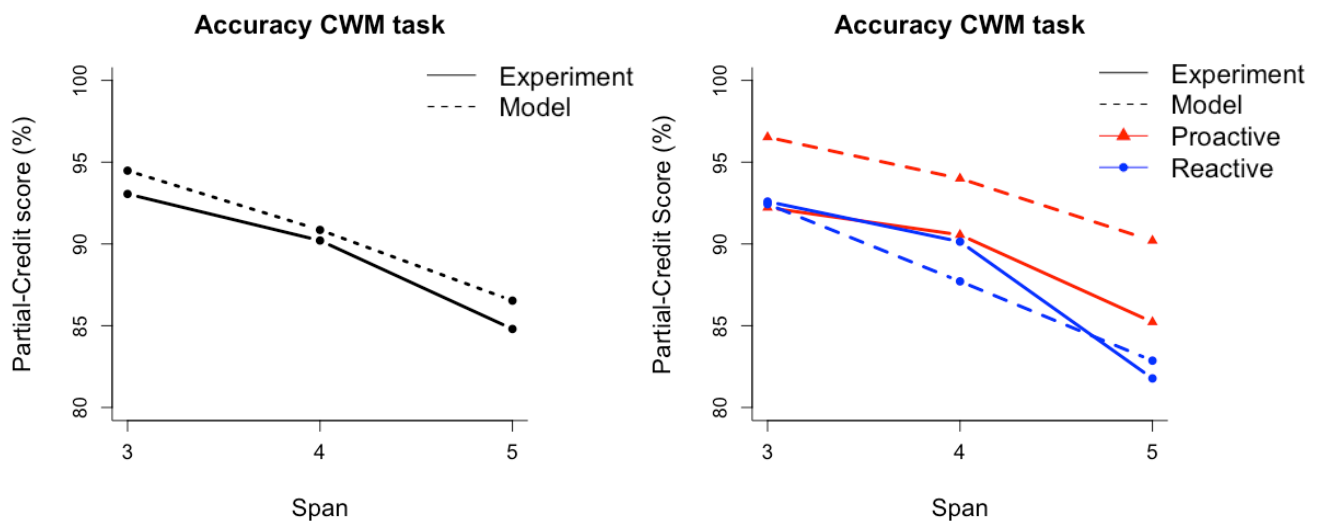


C) Mean error rate on the Names task depicted per control strategy. Proactive control is shown in red and reactive control in blue. The solid line represents the behavioral data and the dotted line the model data.

Figure 21: Comparison of the outcomes of the Names task between the model data and the experimental data. Data from all participants is used in plots A and B, and the data is dichotomized in proactive and reactive control in plot C.

The CWM task

Only accuracy data were used in the behavioral analyses of the CWM task, as participants did not receive any instructions to perform the task rapidly. To compare the model outcomes to those of the behavioral experiments, we calculated the model's average PCS score per letter span. The model results are shown in Figure 22. The model's PCS score per letter span matched the results found in the behavioral data of all participants quite nicely. To recap on the results, the effects of the letter span on the mean PCS score was found to be marginally significant, as were the effects of the proactiveness index on the mean PCS score. When we examined the outcomes of the proactive versus the reactive models, however, we did see a clear distinction between a proactive and a reactive control strategy.



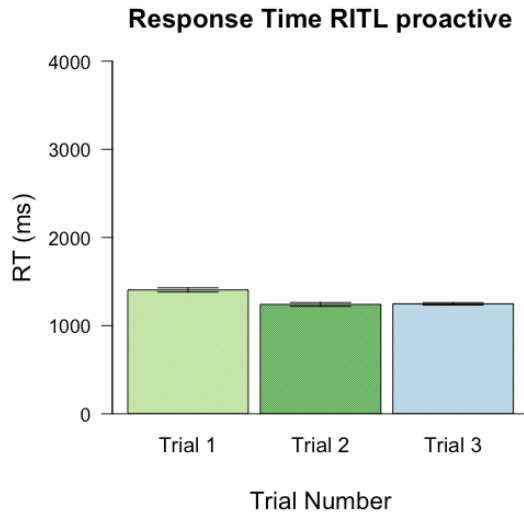
A) Mean partial-credit unit score (PCS) on the storage task of the CWM task depicted for all participants. The solid line represents the behavioral data and the dotted line the model data.

B) Mean partial-credit unit score (PCS) on the storage task of the CWM task depicted per control strategy. Proactive control is shown in red and reactive control in blue. The solid line represents the behavioral data and the dotted line the model data.

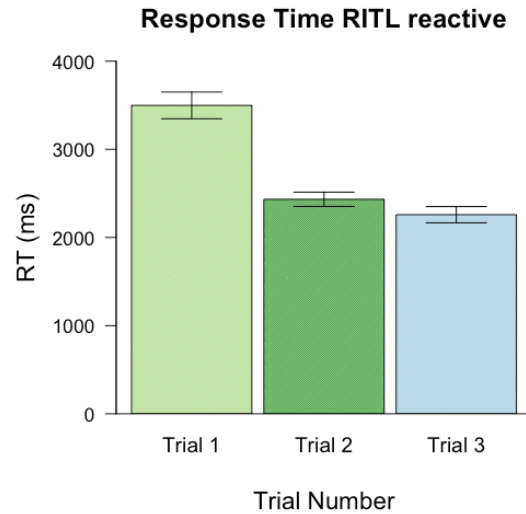
Figure 22: Comparison of the outcomes of the CWM task between the experimental data and the model data. Data from all participants is used in plot A, and the data is dichotomized in proactive and reactive control in plot B.

PRIMs: Proactive vs reactive control

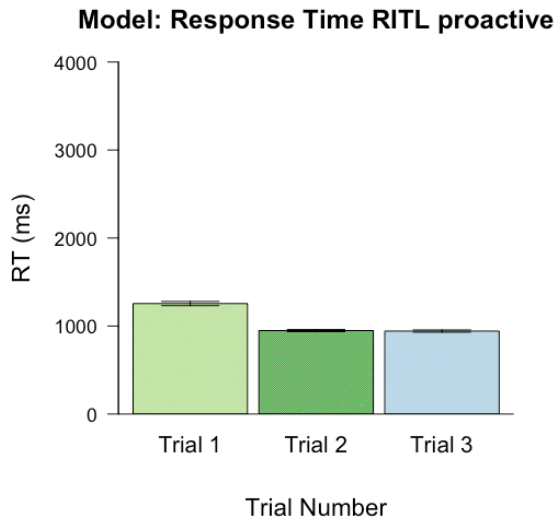
In Figure 7, we saw that participants were slowest to respond in the first trials of the RITL task. In addition, the within-subjects SE was the largest in those trials. Furthermore, Figure 13 and 14 showed us that the distinction between proactive and reactive control was most substantial in the first trials of the RITL task. Figure 23 depicts the model fit of the RTs on the RITL task of proactive and reactive participants in the experimental data (Figure 13 and 14 are replotted in 23A and 23B). The model results match the RTs of proactive and reactive participants in the behavioral data nicely, albeit that both models responded too fast on the second and third trials. Also, the within-subjects SE in the model data was highest in the first trials, which mirrored the results found in the experimental data.



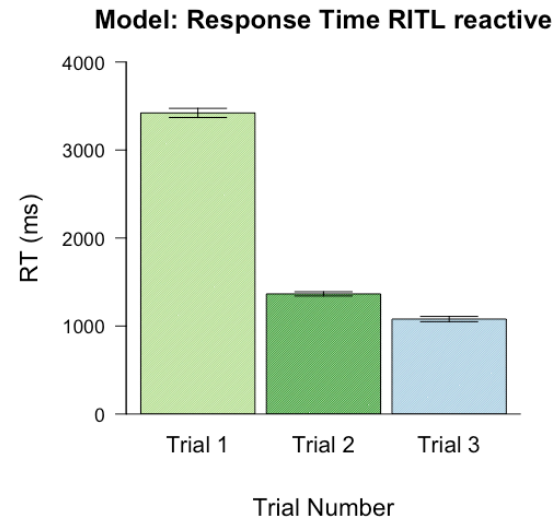
A) Mean RT in milliseconds per trial number (1,2,3) of proactive participants in the behavioral data depicted with the within-subject SE.



B) Mean RT in milliseconds per trial number (1,2,3) of reactive participants in the behavioral data depicted with the within-subject SE.



C) Mean RT in milliseconds per trial number (1,2,3) of the proactive model depicted with the within-subject SE.



D) Mean RT in milliseconds per trial number (1,2,3) of the reactive model depicted with the within-subject SE.

Figure 23: Comparison between the proactiveness index in the experimental data and the model data. The experimental data is depicted in plots A and B. The model data is shown in plots C and D.

Discussion

In the present body of work, we have answered the following research question: In what way does the participant's cognitive control strategy predict the performance across different problem-solving tasks? Three tasks were performed and computational models were developed in PRIMs. We expected participants to benefit from a proactive control strategy on all three tasks, as in all tasks goal-relevant information had to be held active to successfully complete the task. Furthermore, we theorized the proactiveness index (i.e. mean RT on the first trials of the RITL task) to be an indicator of the participants' cognitive control strategies, as fast response times on the RITL task meant for an individual to have prepared for the upcoming stimuli. We formulated three hypotheses: firstly, participants high on the proactiveness index, who we consider proactive, perform more accurately on the RITL task as well as the Names task and the CWM task than participants low on the proactiveness index, who we consider reactive; secondly, our implementation of the two control modes in the computational models explains the variations in performance found in the behavioral data of all three tasks; and lastly, participants with a high accuracy score on the CWM task, who thus are high in working memory capacity, score higher on the proactiveness index than participants with a low accuracy score on the CWM task, who are low in working memory capacity. In the upcoming section, we will examine each of the hypotheses in detail.

In the behavioral experiments, we found that subjects fast on the RITL task – who we considered proactive – scored significantly better on both the RITL task and the Names task than participants who were slow on the RITL task, who thus were considered reactive. Results from the CWM task were marginally significant. These results largely confirmed the first hypothesis, that participants high on the proactiveness index, who we consider proactive, perform more

accurately on the RITL task as well as the Names task and the CWM task than participants low on the proactiveness index, who we consider reactive.

Next, our model results were to fit the behavioral data. The models of both the Names task and the CWM task adopted the strategy to rehearse to exercise proactive control, whereas the model of the RITL task used the strategy to prepare. When the reactive control strategy was applied, all models just waited for the next event to happen. We hypothesized that our implementation of the two control modes in the computational models explains the variations in performance found in the behavioral data of all three tasks. The model results of the RITL task fitted the outcomes found in the behavioral data quite nicely, which posited a first indicator for us to confirm the second hypothesis. The models of the Names task, however, did not support these findings. Whereas the behavioral data indicated there to be a clear distinction between a proactive and a reactive participant, our model data did not. In the CWM task, on the contrary, the distinction between a proactive and a reactive participant was clearly present in the model data but absent in the behavioral data. Altogether, these results did not confirm our second hypothesis. Nevertheless, we have demonstrated the proactiveness index to measure more than the mere difference between a fast and slow performing participant, as the participants' response times on the RITL task were unrelated to their response times on the Names task. Hence, although our results did not confirm the second hypothesis, they still do signal a difference between strategies to exist. A topic we will discuss below.

Lastly, we expected participants high in WMC to be more prone to engage in proactive control than participants low in WMC. As, according to the DMC framework (Braver et al., 2007), the construct to measure WMC is based on the same underlying mechanism as the construct to measure cognitive resources, individuals with a greater WMC would have had more

resources available and thus, should have been able to engage in proactive control more regularly. We studied the last of our hypotheses – participants with a high accuracy score on the CWM task, who thus are high in working memory capacity, score higher on the proactiveness index than participants with a low accuracy score on the CWM task, who are low in working memory capacity – using the outcomes of the behavioral experiments. We were unable, however, to indicate there to be an effect of the participant's working memory capacity on the cognitive control strategy that was adopted. In our experimental set-up, participants could elect themselves which control type to use. Hence, participants high in WMC could still have had more cognitive resources available, and thus should have been able to engage in proactive control more regularly, but in the end, the choice was theirs. All in all, solely the first hypothesis was confirmed by our results. In the next section, we will discuss the implications of the results as well as the limitations of the study.

Implications and limitations

In their study on executive functions, Miyake et al. (2000) distinguished between the three components of cognitive control (i.e. shifting, inhibition & updating), albeit as latent variables. Hence, we know that all three components affect performance differently, but we do not know exactly how. In the present body of work, we examined the dual-mechanisms of control (DMC) framework (Braver et al., 2007), which approaches cognitive control differently; cognitive control is achieved through two different control modes. Whereas proactive control requires internal preparation, reactive control is solely based on external events (Braver, 2012; Braver et al., 2007).

Prior to the experiments, we assumed that rehearsal would explain the variations in performance on the Names task, but we were mistaken. Even after several revisions, we were unable to model the differences between a proactive and a reactive control strategy. We expected participants to fully understand how to proceed with the task when the instruction (e.g. 12) was shown, but that might have been erroneous. In hindsight, the differences in performance between a proactive and a reactive participant might as well have been caused by the participant's preparedness to process the instruction (e.g. 12). This implies that proactive participants would have understood better how to proceed after the instruction was shown and that this caused the variations in performance that we found in the behavioral data. At the onset of the study, we believed rehearsal to be the main strategy for individuals to adopt when exercising proactive control. Our results, however, indicate proactive participants to perform more accurately due to their willingness to prepare for a task. Equal to the RITL task, proactive participants might have benefitted from their preparedness to process task-relevant information in the Names task. Admittedly, in the Names task, the participants' cognitive control strategies did not affect their response times, and for the abovementioned explanation to hold true, one might expect reactive participants to perform the Names task slower than proactive participants. Reactive participants, however, might have felt more time pressure, which then would have translated into additional mistakes instead of an increase in latency. All in all, the participant's preparedness to process task-relevant information (i.e. the instruction) posits an alternative explanation for the distinction between a proactive and a reactive control strategy that we found in the behavioral data of the Names task, and constitutes an interesting topic for future research.

In the CWM task, the effects of the participants' control modes were found to be marginally significant. In both the Names task and the RITL task, participants had to process

newly acquired information, while in the CWM task participants simply had to memorize letters. The CWM task might have been too easy, in that participants who adopted a reactive control strategy still performed well. As a result, the differences in performance between proactive and reactive participants were simply too small for us to detect.

In this work, we theorized the participant's mean RT on the first trials of the RITL task to be an indication of the participant's cognitive control strategy, and we used that as our "proactiveness index". Figure 13 and 14 confirmed the first trials of the RITL task to be the best indicator for the proactiveness index. The model results of the RITL task, in which the strategy to prepare was used to simulate proactive control, fitted the behavioral data nicely, specifically the RTs on the first trials of proactive and reactive participants (Figure 23). Both models, however, responded too fast on the second and third trials of the RITL task. In the current implementation of the RITL task (Figure 17), the model started over at "apply the instructions to the stimuli", when it had provided its answer for the first trial and was ready to proceed to the next (i.e. the second & third trials). Thus, we assumed participants to have perfect recall of the instructions and their implications after the first trial of a task. An oversimplification, as we only used the first trials on the RITL task in our comparisons between the model results and the outcomes of the behavioral experiments. Of course, these results might indicate the participants' control strategies to be only part of the explanation of the differences that we found in the behavioral data, but it is more likely that our implementation of the control modes did not cover all that constitutes proactive versus reactive control in the second and third trials of the RITL task.

The computational models, which were developed in the PRIMs (Taatgen, 2013b) cognitive architecture, accounted for many of the effects that we found in the behavioral data.

Nevertheless, we encountered some limitations. Firstly, PRIMs offers few constraints on how to model a task. To give an example, PRIMs provides no constraints on the way new knowledge is stored in declarative memory or on the way the model should prepare for a response. As a result, too much variability existed in the way the tasks could be implemented. Secondly, our models only allowed for retrieval errors, whereas other types of errors (e.g. transposition errors) were unaccounted for. Lastly, we were unable to implement the distinction between novel and repeated trials in our models of the RITL task, as the version of PRIMs in which we started to develop our models did not yet support a more hierarchical representation of working memory.

As we aimed to compare the model results with the outcomes of the behavioral experiments, we dichotomized the behavioral data into two groups. Dichotomizing, however, comes at a price. First and foremost, data is lost. In the analyses of the two control modes in our computational models, we discarded data from 20 participants. In addition, participants who were close to the cut point, but on opposite sides, were deemed very different, rather than quite the same (see Altman & Royston (2006) for an elaborate analysis on dichotomizing continuous variables). This constitutes a topic for future research, see below.

Future research

In the “Discussion” section, we already briefly mentioned some topics for future research. First and foremost, we might have misinterpreted the main strategy that participants adopt when exercising proactive control. Instead of rehearsal, the participant’s preparedness to process task-relevant information might have caused the differences in performance found in the behavioral data of both the Names task and the RITL task. Further development of the computational models of the Names task in PRIMs is warranted to examine this hypothesis. Next, we dichotomized the behavioral data, as we created our computational models to be either extremely proactive or extremely reactive. In real life, participants are somewhere in between. In a future experiment, both control modes could be combined in one computational model for each task, wherein the degree of proactiveness (i.e. the chance of acting proactive or just waiting) varies per run and is established by means of the participant’s proactiveness index. As a result, the degree of proactiveness of the computational models will be on a continuum, instead of two extremes. Consequently, the behavioral data will not have to be dichotomized and therefore, no data will be lost. In subsequent data analysis, one could perform mixed-effects linear and mixed-effects logistic regression analyses on both the model data and the behavioral data and compare the results. Also, we did not find an effect of working memory capacity on the participants’ cognitive control strategies, potentially caused by the participants’ freedom to elect themselves which control type to adopt during the experiments. If one pursues to study the effects of working memory capacity on the participants’ cognitive control strategies, in a next experiment, subjects should be instructed to adopt a proactive strategy. Lastly, in our models of the RITL task, we assumed subjects to recall the instructions and their implications perfectly in the second and third trials of a task. Further development of the computational models is needed, to assess

whether the models only responded too fast because of the way we made them proceed after the first trials. Alternatively, both control modes might have only explained parts of the differences that we found in the behavioral data. As proactive control requires goal-relevant information to be held active, and reactive control incorporates goal-irrelevant processing (Braver, 2012; Braver et al., 2007), the instructions and their implications could, for example, be forgotten more regularly at the onset of the second and third trials in the reactive model of the RITL task. Concurrently, one could implement the distinction between novel and repeated trials in the models of the RITL task, as the current version of PRIMs does support more hierarchical representations of working memory.

Conclusion

In this thesis, we have answered the following research question: In what way does the participant's cognitive control strategy predict the performance across different problem-solving tasks? We theorized working memory control to be task-general, and examined working memory control using the dual-mechanisms of control framework. We found that participant who responded fast on the first trials of the RITL task, who were considered proactive, performed better on both the RITL task and the Names task than participants who responded slow on the first trials of the RITL task, who we considered reactive. Furthermore, we demonstrated the proactiveness index to measure more than the mere difference between fast and slow performing participants. The model results, however, did not confirm the variations in performance to be caused by the participants' control strategies across different problem-solving tasks. Only in the RITL task, the model results fitted the behavioral data nicely. Two explanations were proposed, which are not mutually exclusive. Either individuals mainly adopt proactive control with the strategy to prepare rather than the strategy to rehearse, or the two control modes of the DMC framework did explain only parts of the variations in performance that we found in the behavioral data. All in all, initial steps were taken to improve our understandings of what constitutes working memory control.

References

- Altman, D. G., & Royston, P. (2006). The cost of dichotomising continuous variables. *BMJ*, 332(7549), 1080. <https://doi.org/10.1136/bmj.332.7549.1080>
- Anderson, J. R. (2007). *How Can the Human Mind Occur in the Physical Universe?* New York: Oxford University Press.
- Anderson, J. R., Qin, Y., Jung, K. J., & Carter, C. S. (2007). Information-processing modules and their relative modality specificity. *Cognitive Psychology*, 54(3), 185–217. <https://doi.org/10.1016/j.cogpsych.2006.06.003>
- Bell, C. G., & Newell, A. (1971). *Computer Structures: Readings And Examples*. New York: McGraw-Hill.
- Braver, T. S. (2012). The variable nature of cognitive control: A dual-mechanisms framework. *Trends in Cognitive Sciences*, 16(2), 106–113.
- Braver, T. S., Gray, J. R., & Burgess, G. C. (2007). Explaining the Many Varieties of Working Memory Variation: Dual Mechanisms of Cognitive Control. *Variation in Working Memory*, 76–106. <https://doi.org/10.1093/acprof:oso/9780195168648.001.0001>
- Chein, J. M., & Morrison, A. B. (2010). Expanding the mind's workspace: training and transfer effects with a complex working memory span task. *Psychonomic Bulletin & Review*, 17(2), 193–199. <https://doi.org/10.3758/PBR.17.2.193>
- Cole, M. W., Bagic, A., Kass, R., & Schneider, W. (2010). Prefrontal Dynamics Underlying Rapid Instructed Task Learning Reverse with Practice. *Journal of Neuroscience*, 30(42), 14245–14254. <https://doi.org/10.1523/JNEUROSCI.1662-10.2010>
- Conway, A. R., Kane, M. J., Bunting, M. F., Hambrick, D. Z., Wilhelm, O., & Engle, R. W. (2015). Working memory span tasks: A methodological review and user's guide.

- Psychonomic Bulletin & Review*, 12(5), 769–786.
- Cowan, N. (2000). Processing limits of selective attention and working memory: Potential implications for interpreting. *Interpreting*, 5(2), 117–146.
- Cowan, N., & Vergauwe, E. (2014). Applying how adults rehearse to understand how rehearsal may develop. *Frontiers in Psychology*, 5, 2014–2016.
- Daamen, J. (2016). *Distracted by the self*. Unpublished paper, Department of Artificial Intelligence, University of Groningen, Groningen, The Netherlands.
- Dehaene, S., Erszberg, M., & Changeux, J. P. (1998). A neuronal model of a global workspace in effortful cognitive tasks. *Proceedings of the National Academy of Sciences*, 95(24), 14529–14534.
- Dehaene, S., Meyniel, F., Wacongne, C., Wang, L., & Pallier, C. (2015). The Neural Representation of Sequences: From Transition Probabilities to Algebraic Patterns and Linguistic Trees. *Neuron*, 88(1), 2–19. <https://doi.org/10.1016/j.neuron.2015.09.019>
- Kane, M. J., & Engle, R. W. (2002). The role of prefrontal cortex in working-memory capacity, executive attention, and general fluid intelligence: An individual-differences perspective. *Psychonomic Bulletin & Review*, 9(4), 637–671. <https://doi.org/10.3758/BF03196323>
- Mathôt, S., Schreij, D., & Theeuwes, J. (2012). OpenSesame: an open-source, graphical experiment builder for the social sciences. *Behavior Research Methods*, 44(2), 314–324. <https://doi.org/10.3758/s13428-011-0168-7>
- Miyake, A., Friedman, N. P., Emerson, M. J., Witzki, A. H., Howerter, A., & Wager, T. D. (2000). The Unity and Diversity of Executive Functions and Their Contributions to Complex “Frontal Lobe” Tasks: A Latent Variable Analysis. *Cognitive Psychology*, 41(1), 49–100. <https://doi.org/10.1006/cogp.1999.0734>

- Newell, A. (1973). You can't play 20 questions with nature and win: Projective comments on the papers of this symposium. *Visual Information Processing*, 283–308.
<https://doi.org/10.1021/bi047419c>
- R Core Team. (2015). R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria. Retrieved from <http://www.r-project.org/>
- Satterthwaite, F. E. (1946). An approximate distribution of estimates of variance components. *Biometrics bulletin*, 2(6), 110-114.
- Singley, M. K., & Anderson, J. R. (1985). The transfer of text-editing skill. *International Journal of Man-Machine Studies*, 22(4), 403–423. [https://doi.org/10.1016/S0020-7373\(85\)80047-X](https://doi.org/10.1016/S0020-7373(85)80047-X)
- Stocco, A., Lebiere, C., & Anderson, J. R. (2010). Conditional Routing of Information to the Cortex: A Model of the Basal Ganglia's Role in Cognitive Coordination. *Psychological Review*, 117(2), 541–574. <https://doi.org/10.1037/a0019077>.Conditional
- Taatgen, N. A. (2013a). Diminishing return in transfer: A PRIM model of the Frensch (1991) arithmetic experiment. *Proceedings of the International Conference on Cognitive Modeling*, 29–34.
- Taatgen, N. A. (2013b). The Nature and Transfer of Cognitive Skills, 120(3), 439–471.
<https://doi.org/10.1037/a0033138>
- Taatgen, N. A., Katidioti, I., Borst, J., & Van Vugt, M. (2015). A Model of Distraction using new Architectural Mechanisms to Manage Multiple Goals. *ICCM Conference*, 264–269.

Appendix A: Questionnaire

Demographics
Subject number
Age
Gender
Primary language
Education programme
Did you perform one of these tasks before? And if so, which one?
RITL task
I tried to actively rehearse the instructions right after the third task rule was presented
I waited until the two stimuli appeared on the screen before trying to remember the instructions
I did notice the distinction between the bold and thin borders of the tasks presented on the screen
I started to integrate the instructions right after the third task rule was presented
I waited until the two stimuli appeared on the screen before integrating the instructions
I got distracted in between the instructions and the two stimuli
CWM task
I tried to focus on the shoebox questions of the processing task
I tried to focus on memorizing the letters in the storage task
I tried to actively rehearse the letters that were presented in between both tasks
I waited until the recall screen before trying to remember the letters that were presented

I got distracted while performing both tasks
NAMES task
I did manage to learn the word-number associations
I tried to actively rehearse the order of names after they were introduced
I waited until the instruction appeared before trying to remember the order of names
I tried to rehearse the word-number associations throughout the task
I got distracted between the presentation of the third name and the instruction

Appendix B: Stimuli RITL task

Old stimulus	New stimulus
Clover	Frog
Fern	Pepper
Fungus	Spinach
Iguana	Shrek
Ivy	Pistachio
Mucus	Basil
Shamrock	Artichoke
Shrub	Cabbage
Vine	Parsley
Tangerine	Lolly pop
Starfruit	Sugar cube
Honeysuckle	Sweet potato
Honeydew	Popcorn
Cantaloupe	Icecake
Yarn	Pillow
Quilt	Jelly bean
Gelato	Cotton candy
Cymbals	Train
Gumdrop	Gummy bear
Parfait	Snickers

Slippers	Silk
Mint	Christmas tree

Appendix C: Instructions RITL task

The goal of the next experiment is to quickly read and apply instructions for many small tasks. In each task, you will be presented with new instructions that you have to memorize and apply to two stimuli.

A complete task looks like this:



You can respond using the keyboard with the buttons 2 (left middle finger), 3 (left index finger), 8 (right index finger) and 9 (right middle finger). It is imperative that you **always keep both hands on the keyboard**.

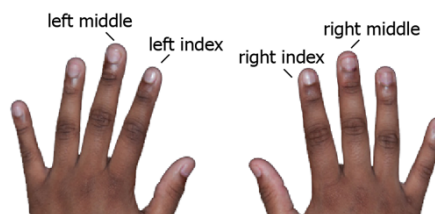
Each instruction consists of three rules.

An example is provided for each of the different rules.

Please read these carefully:

The combination **green; just one; right index** means:

If the answer to the question “**green?**” is true for **just one** of both stimuli, press your **right index** finger. In case *both* objects are green or *neither* of the objects is green you respond *no* by pressing your **right middle** finger.



The combination of **sweet; same; left middle** means:

If the answer to the question “**sweet?**” is the **same** for both stimuli, press your **left middle** finger. Thus, you respond yes when both stimuli are either sweet or not sweet; note that when both stimuli are not sweet, you respond yes as well! You respond *no* (**left index**) when *just one* of the stimuli is sweet.

The combination of **loud; second; right middle** means:

If the answer to the question “**loud?**” is true for the **second** stimulus, press your **right middle** finger. In case the second stimulus is *not* loud you respond *no* by pressing the **right index** finger. Note that the first stimulus is irrelevant in this scenario.

The combination of **soft; not second; left index** means:

If the answer to the question “**soft?**” is **not** true for the **second** stimulus, press your **left index** finger. In case the second stimulus *is* soft, you respond *no* by pressing the **left middle** finger. Again, note that the first stimulus is irrelevant in this case.

First, you are going to practice. Then, you will perform the test. Throughout the test, novel tasks are introduced in addition to the four tasks that were practiced. The tasks that were practiced will have **thick** borders, while the novel tasks have **thin** borders.

Try to execute the tasks as quickly and accurately as possible. Please contact the experimenter once you feel ready to start.

Good luck!

Appendix D: Stimuli CWM task

Word	Translation	Does the object fit in a shoebox?
geld	money	yes
foto	picture	yes
papier	paper	yes
brief	letter	yes
cd	CD	yes
handvat	knob	yes
vinger	finger	yes
suiker	sugar	yes
kogel	bullet	yes
poeder	powder	yes
boter	butter	yes
pistool	pistol	yes
honing	honey	yes
cent	penny	yes
kristal	crystal	yes
sinaasappel	orange	yes
knikker	cabbage	yes
kaars	candle	yes
citroen	lemon	yes

Word	Translation	Does the object fit in a shoebox?
halsband	collar	yes
snoep	candy	yes
stof	dust	yes
naald	needle	yes
ui	onion	yes
insekt	bug	yes
kompas	compass	yes
peper	pepper	yes
zeepbel	soap bubble	yes
lint	ribbon	yes
knoop	button	yes
parfum	perfume	yes
boterham	sandwich	yes
puzzel	puzzle	yes
appel	apple	yes
bes	berry	yes
hamer	hammer	yes
salade	salad	yes
diamand	diamond	yes
kers	cherry	yes
veer	feather	yes

Word	Translation	Does the object fit in a shoebox?
oester	oyster	yes
katje	kitten	yes
olijf	olive	yes
roodborstje	parakeet	yes
spin	spider	yes
juweel	jewel	yes
lelie	lily	yes
eekhoorn	squirrel	yes
panty	tights	yes
mus	sparrow	yes
stad	city	no
mens	human	no
lichaam	body	no
universiteit	university	no
tafel	table	no
eiland	island	no
rivier	river	no
gebouw	building	no
markt	market	no
hotel	hotel	no
station	station	no

Word	Translation	Does the object fit in a shoebox?
keuken	kitchen	no
dorp	village	no
pilaar	pillar	no
bos	forest	no
geweer	gun	no
tuin	garden	no
motor	engine	no
wagen	car	no
slaapkamer	bedroom	no
bureau	desk	no
snelweg	highway	no
paleis	palace	no
tempel	temple	no
haven	harbor	no
voetbal	football	no
oceaan	ocean	no
berg	mountain	no
deken	blanket	no
kelder	basement	no
hut	cabin	no
planeet	planet	no

Word	Translation	Does the object fit in a shoebox?
kapel	chapel	no
fontein	fountain	no
mand	basket	no
leeuw	lion	no
weide	grassland	no
kast	closet	no
anker	anchor	no
deur	door	no
tapijt	carpet	no
koets	coach	no
harnas	harness	no
pony	pony	no
kalkoen	turkey	no
kasteel	castle	no
schoorsteen	chimney	no
wieg	cradle	no
oven	furnace	no
tijger	tiger	no