UNIVERSITY OF GRONINGEN

THESIS HUMAN-MACHINE COMMUNICATION

Bilinguals Already Know: Reducing Internal Control Leads to Faster Task-Switching Performance.

Author:
Charlotte DE BLECOURT
(s2188813)

Supervisors:
dr. Jelmer BORST
dr. Andrea STOCCO

July 16, 2018

rijksuniversiteit
groningen

Contents

Abstract

Rapid Instructed Task Learning (RITL) involves rapid switching between instructions and stimuli. In a previous RITL experiment by Stocco and Prat (2014), bilingual and monolingual participants solved sets of arithmetic operations. Monolinguals executed novel sets ± 500 ms slower than practiced sets, while bilinguals executed both novel and practiced sets at the same pace. In other words, bilinguals may not have changed their execution strategy; they may already know how to efficiently switch between task instructions and information. Bilingualism increases an individual's linguistic demands. Bilinguals may therefore reduce internal control in exchange for faster information routing between cortical areas, resulting in more fluent language use at the expense of accuracy. This effect might spill over in non-linguistic task-switching, thus resulting in lower reaction times in task-switching paradigms. This hypothesis is in line with Abutalebi and Green's adaptive control hypothesis and Stocco et al.'s conditional routing model. To investigate this hypothesis, we looked at the computational steps that bilinguals and monolinguals take during a RITL trial. We created computational models of monolinguals and bilinguals for the RITL paradigm as seen in Stocco and Prat (2014), using the Adaptive Control of Thought–Rational (ACT–R) architecture. This allows us to observe RITL task performance as an adaptive step-by-step process. The "monolingual" model performs each step in the task separately. With experience, these separate steps merge into habitual sequences of actions, in which multiple computations are performed at once. The "bilingual" model has these compiled sequences by design, and internal control checks have been removed. This reduces execution time and eliminates learning effects. The models could not reproduce the exact reaction times, but they correctly reproduced the behavioral pattern observed in Stocco and Prat over a wide parameter space. These results suggest that the additional linguistic demands of bilingualism are met by reducing internal control in exchange for faster information routing between cortical areas, and that this effect carries over to non-linguistic rule-switching as well.

Keywords: ACT-R; Rapid Instructed Learning Task; Bilingualism

1 Introduction

Over 86% of the Dutch (Eurostat, 2018) and about a quarter of the American population (United States Census Bureau, 2013) speak at least one other language besides their native tongue. Increasing evidence suggests that multilingualism has additional benefits besides the ability to communicate outside one's native language. Bilinguals tend to outperform monolinguals in cognitive control tasks (e.g. (Bialystok, Craik, Klein, & Viswanathan, 2004; Bialystok, 2009)), especially in task-switching paradigms that are designed to measure executive function, such as the Simon task (Bialystok et al., 2004; Prior & MacWhinney, 2010).

Executive function is an overarching name for activities that involve inhibition, shifting, and updating (see for example (Miyake et al., 2000)). Bilingualism seems to train the latter two (Prior & MacWhinney, 2010; Carlson & Meltzoff, 2008). As both languages are always active and competing (MacWhinney, 1997), bilinguals constantly need to control the flow of information. This "training" improves the efficiency of task-switching, which is in turn observed as faster reaction times in task-switching paradigms (Prior & MacWhinney, 2010). This supports the idea that task-switching is less taxing for bilinguals (Stocco, Yamasaki, Natalenko, & Prat, 2014). Moreover, neuroimaging findings suggest that the bilingual experience trains specific brain circuits involved in flexible rule selection and application (Stocco & Prat, 2014; Stocco et al., 2014), which are an important aspect of executive function. The bilingual experience may therefore result in computational advantages that spill over in non-linguistic tasks. However, although the behavioral and neural effects have been studied, the mechanisms behind the computational advantages of bilingualism are unknown.

In the present paper, I will explain the computational advantage of bilingualism as exchanging internal control for faster information routing. First, I will explain the current theories behind the bilingual advantages in task-switching. Next, I will explain the advantage of using the Rapid Instructed Task Learning paradigm to investigate task-switching behavior. These theories become the foundation of the bilingual and monolingual ACT-R models of Stocco and Prat's (2014) RITL task. The models differ in two ways. The monolingual model performs ACT-R buffer manipulations separately and checks buffer availability before executing the task. The model is able to merge computational steps (known as *production rules*), and removes buffer queries in the process. The bilingual model already has these merged production rules and has no queries by design. These principles reduce response times, but may result in a lower accuracy.

1.1 Current Models of Bilingual Task-Switching

In this section, I will outline the current theories behind the bilingual advantages in task-switching. The *conditional routing model* (Stocco, Lebiere, & Anderson, 2010; Stocco et al., 2014) suggests that signals usually travel over the cortex. When the regular cortico-cortical route cannot sufficiently keep up with external demands - our case, linguistic demands - , an alternative neural route will be sought through the basal ganglia. The other theory, Green and Abutalebi's cognitive control hypothesis (Green & Abutalebi, 2013), states that language control processes themselves adapt to demands from the environment.

1.1.1 The conditional routing model

According to the conditional routing model (Stocco et al., 2010, 2014), the basal ganglia re-route signals between cortical regions. Without basal ganglia intervention, the flow of signals across cortical networks is determined by the strength of cortico-cortical projections, which are shaped by practice. The basal ganglia step in to actively shape behavior by prioritizing different signals and overriding preexisting cortico-cortical connections. With practice, intermediate steps through the basal ganglia are eliminated. This learning process results in a stable trade-off between task speed and automaticity.

In language, the conditional routing model comes in to play when linguistic rules increase in complexity (Stocco et al., 2014; Seo, Stocco, & Prat, 2018). For example, besides the regular conjugation of a verb, the English language has numerous irregular verbs. This challenges the established cortico-cortical pathways, because they require flexible activation of the right pathways under the right conditions. The rules and requirements get permanently stored with enough practice. The conditional routing model also connects the relationship between language and executive function: although they are two different aspects of cognitive functioning, they rely on the same set of computations through the basal ganglia circuitry.

Bilingualism provides an individual with a whole new set of rules that are being used separately from the other language. This increases demands on the basal ganglia to select appropriate rules and representations, and to switch between rules and representations depending on the intended language. Thus, second language proficiency may result in the side effect of increasing the ability of the basal ganglia to exert control over established cortico-cortical connections. Findings indeed suggest that bilinguals and monolinguals differ in re-routing when task demands are increasing. For example, bilinguals are more effectively modulating basal ganglia activity when faced with changing task demands (Stocco & Prat, 2014). However, monolinguals may learn alternative routes during task execution, which improves their performance over time when faced with increasing demands for a specific task.

1.1.2 The adaptive control hypothesis

The adaptive control hypothesis (Green & Abutalebi, 2013) takes another approach by asserting that the demands of language control processes in bilingual speakers are higher than in monolingual speakers. Second language usage highly depends on context. For example, one might use two different languages at work (*dual language context*), or one language at work and one at home (*single language context*). As the context of the language usage changes, the demands for control of cognitive processes change along too. Each interactional context affects control processes in a different way. In particular, goal maintenance, interference control, response inhibition, and task (dis-)engagement are affected in single and/or dual language contexts (Abutalebi & Green, 2007; Green & Abutalebi, 2013).

The language control processes are tied to a wide network of brain regions (Green & Abutalebi, 2013; Abutalebi & Green, 2016). For the adaptive control hypothesis, these regions are divided into the *speech pipeline* and the *control network*. The basal ganglia are placed as a 'gate' that control the information flow between the prefrontal cortex and posterior cortical regions (Crinion et al., 2006; Abutalebi & Green, 2016). The anterior cingulate cortex (ACC) and the presupplementary motor area (pre-SMA) are particularly important in conflict monitoring and initiating speech in language switching (Luk, Green, Abutalebi, & Grady, 2012).

The non-verbal advantage of bilinguals could be a result of using control processes more often in a verbal context (Green & Abutalebi, 2013). Context-dependent control is exercised by a common mechanism, which is also active in non-linguistic situations. Thus, in situations where additional adaptive control is needed, bilinguals exhibit better adaptive responses compared to monolinguals, such as lower switch-costs (Prior & MacWhinney, 2010).

1.2 Rapid Instructed Task Learning

Although task-switching paradigms have shown a general advantage of bilingualism, they often cannot show where the computational advantage is. Tasks may not distinguish between executive function components, or even the difference between task understanding and actual execution of instructions. This division is made in Rapid Instructed Task Learning (RITL; Cole, Laurent, and Stocco (2013); Stocco, Lebiere, O'Reilly, and Anderson (2012)), which involves high-paced task learning and execution. Figure 1 shows an example RITL trial. Participants are presented with sets of instructions, a set of stimuli, and perhaps a probe. Each trial, the participant applies the set of instructions to the stimuli and answers accordingly. Although the main goal of the experiment remains the same, the instructions and stimuli change with each trial. This forces participants to quickly reconfigure their "mental template" of the task: they must apply new information to new instructions with every trial.

The trials can be separated into three phases. In the *encoding* phase, instructions are presented on the screen and remembered for further use. Instructions are familiar tasks, such as arithmetics and go no-go tasks. The actual stimuli are presented in the *execution* phase. The instructions are applied to the stimuli. Next, the subject continues to the *response* phase, and communicates their answer. To separate the execution and response phases, participants may need to enter their answer in a separate screen (such as in Figure 2, which includes a probe). A new trial starts after the response: the instructions change and the mental process starts over again.

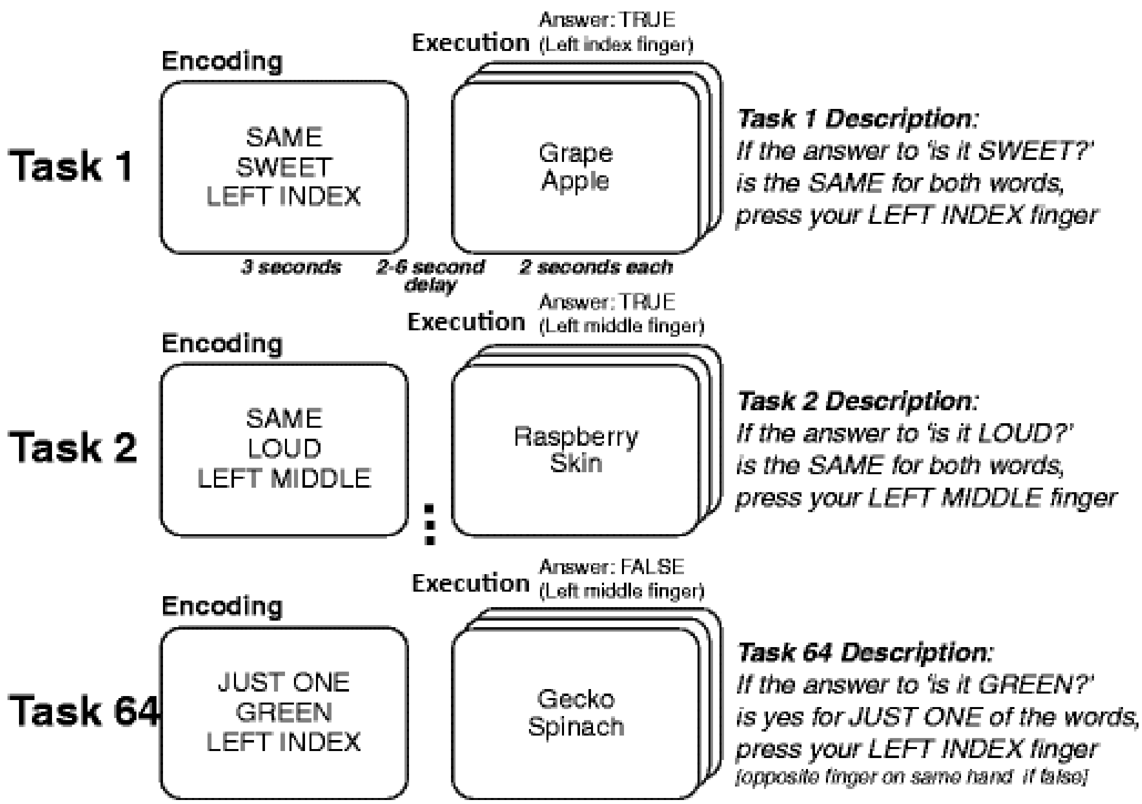

Figure 1: Example RITL paradigm as seen in Cole, Bagic, Kass, & Schneider (2010). Participants performed 64 repeated tasks by re-combining 12 rules into a multitude of unique sets, thus preserving task novelty. Each task consists of a logic rule (e.g. SAME), a sensory semantic rule (e.g. SWEET), and a response rule (e.g. respond with LEFT INDEX finger). In other words, participants use the left index finger when both stimuli are sweet. There were four possible rules of each type. These rules were presented during the *encoding* phase and applied during the *execution* phase. Extracted from Cole, Laurent, & Stocco (2013).

1.2.1 Bilingual and Monolingual Performance on the RITL Task

Bilingual individuals require to choose between multiple rules to achieve the same goal. For example, the standard rule for pluralization in English is to attach an -s to the word: *human* becomes *humans*, *car* becomes *cars*, et cetera. However, in Dutch the suffix might be -en (*mens-mensen*) or -'s (*auto-auto's*). As bilinguals have multiple sets of linguistic rules, they are forced to flexibly maneuver between linguistic rule application, which supposedly gives rise to the greater flexibility in task-switching behaviors when faced with novel or changing rules (Prior & MacWhinney, 2010).

RITL paradigms may a helpful tool to study the bilingual advantage, as they focus on rule application. They separate the memorization of rules from the application. Furthermore, because the exact rules and variables changes with each trial, RITL paradigms focus more on task goals than content. This is similar to grammatical rule application: the overarching goal may remain the same, but the exact words and rules by which this is achieved differ within and between languages.

With this in mind, Stocco and Prat (2014) used a RITL paradigm to investigate the cognitive and neural mechanisms behind flexible rule application in bilinguals. Bilinguals and monolinguals performed similarly on practiced tasks, but bilinguals were faster than monolinguals at executing novel tasks (see Figure 3). This finding supports the hypothesis that the bilingual advantage in executive functions is related to a better ability reconfigure rule-based behavior.

1.3 Hypothesis

In conclusion, the difference between bilinguals and monolinguals on executive function tasks is especially visible in task-switching paradigms (Prior & MacWhinney, 2010). The behavioral results of Stocco and Prat (see Figure 3) suggest that the flexible rule application is better developed in bilinguals. Apparently, monolinguals had to learn a computational strategy that bilinguals may already be utilizing. This hypothesis also arises from both the conditional routing model and the adaptive control hypothesis described above. Monolinguals might need to exert more control when switching between RITL-rules and information. However, this mechanism would already be established in bilinguals, as they are used to switching between linguistic rules.

2 Models

Bilinguals and monolinguals may initially perform Rapid Instructed Task Learning paradigms differently, but monolinguals adapt their strategy to the demands of the execution phase of the task (see Figure 3). A ”bilingual” model and a ”monolingual” model are presented in this section to show how these changes might come to be. The goal of the models is to show that the bilingual advantage observed in RITL-paradigms - and to a broader extend, other task-switching paradigms - is the result of a learned rule application strategy. The computational model should therefore 1) have a learning component, 2) a flexible control mechanism, and 3) be applicable to other paradigms as well.

The full model code and simulation data can be found on `https://github.com/UWCCDL/ACTR_RITL`.

2.1 Task Description

The task is based on the RITL procedures presented in Stocco and Prat (2014). Each trial consists of three parts (see Figure 2). Instructions are a combination of three operations, for example ”DOUBLE”, ”HALF” and ”ADD”. The first two instructions are unary calculations (”DOUBLE”, ”TRIPLE”, ”INCREMENT”, ”DECREMENT”, ”THIRD”, or ”HALF”) and need to be applied to X and Y respectively. The last operation is a binary operation (”TIMES”, ”DIVIDE”, ”ADD”, or ”SUBTRACT”), and applies to the result of the two previous calculations. The stimuli presented in the execution phase are two numbers between 1 and 9. The solution will always be between 0 and 50. Finally, a probe is presented. The participant then indicates if the probe number corresponds to the solution.

Stocco and Prat (2014) presented eighty trials to each participant. Participants performed forty trials to familiarize themselves with the paradigm. The instructions and input numbers differed per trial, but could be duplicates. During a second session, participants were presented with twenty previously trained instruction sets and twenty novel combinations. The task has been slightly modified for the models. They are first presented with twenty training trials; they did not process any distractors. The practiced instructions were always identical (INCREMENT DOUBLE DIVIDE and INCREMENT DOUBLE DIVIDE), but the X and Y stimuli differed. The testing phase consisted of forty trials; twenty previously practiced instructions, and twenty novel combinations.

2.2 Model Implementation

2.2.1 Overview of the ACT-R architecture

The behavior on the RITL task was modeled in the the Adaptive Control of Thought-Rational (ACT-R) cognitive architecture (Anderson et al., 2004; Anderson, 2009) (see also `http://act-r.psy.cmu.edu/`). In the ACT-R architecture, cognition emerges through interaction between modules. For example, in a RITL task, the visual module processes the presented instructions, the retrieval module retrieves relevant mathematical information, the imaginal module manages the intermediate answers, et cetera. They work independently from each other, but can interact through buffers. Buffers can only contain one buffer at the time, which creates a bottleneck effect. Thus, the imaginal buffer may accept information from the visual buffer to adapts its current content, but needs to remove its current content first. The interaction between these modules is managed by the procedural buffer, which executes *production rules*. These rules are programmable IF-THEN statements, than are matched against buffer content. For example, IF the visual buffer contains an instruction set, THEN put the corresponding behavior in the retrieval buffer. Each buffer is mapped to brain areas associated to their task (Borst & Anderson, 2015; Anderson, Fincham, Qin, & Stocco, 2008). The procedural buffer is associated with the caudate nucleus in the basal ganglia.

ACT-R models have declarative knowledge (such as arithmetic facts) and procedural knowledge (mental operations) (Anderson, 2009). Declarative knowledge is represented as clusters (called *chunks*) of smaller features (called *slots*). For example, the presented RITL instruction set is one chuck, with three slots: each for every instruction. The procedural knowledge consists of the available set of production rules. When multiple rules follow each other, they may merge together into one. For example, if one production requires information in the retrieval buffer and another production requires information in the visual buffer before any action is taken, they may merge into one production with both the retrieval AND visual buffer as conditions. This process, called *production compilation* (N. A. Taatgen & Lee, 2003), eliminates intermediate steps in the model, making the process faster over time.

2.2.2 Model foundation

For both models, the RITL-instructions presented to a model (for example, DOUBLE HALF ADD) are one chunk with three slots. The declarative memory contained predefined chunks: ten arithmetic operations, and 3778 facts about arithmetic operations rounded down to the nearest integer.

Figure 2 shows how the model flows through a trial. The ”monolingual” model

Table 1: Model goals during the execution phase of the monolingual model at the start, after the learning phase and for the bilingual model. Although the bilingual model was capable of learning, the model's productions did not change over time.

Monolingual Model	Monolingual Model After Learning	Bilingual Model
execution-x	execution-x	execution-x
retrieve-task-y		
execution-y	execution-y	execution-y
update-scratchpad-y	update-scratchpad-y	update-scratchpad-y
retrieve-task-binary		
execution-binary	execution-binary	execution-binary
update-scratchpad-binary	update-scratchpad-binary	update-scratchpad-binary
done	done	done

separates rule application and variable binding. These may merge into one action by allowing production compilation: this is the learning component, which allows for easier switching between RITL rules and variables. See Table 1, where the models' intermediate goals during the execution phase are displayed. With production compilation internal checks (buffer queries) are sometimes removed, which speeds up the process, creating a more flexible control mechanism. In theory, this creates the possibility of empty or unavailable buffers, leaving more room for errors during trials. This is not the case nor the goal of this model; the model goals have been strictly defined and guide the models through the process. The production rules of "bilingual" model for the execution phase are based on the monolingual model's rules after production compilation: rule application and variable binding may happen simultaneously. For example, the production rule *update-scratchpad-y-start-binary* remembers the solution to the second arithmetic problem while retrieving the third remain problem. In other words, the bilingual model has already learned what the monolingual model can learn during task execution. The LISP-code can be found on `https://github.com/UWCCDL/ACTR_RITL`. A comparison between the bilingual and monolingual models' productions in the execution phase can be found in Appendix A.

2.2.3 Encoding phase

The encoding phases of the monolingual and bilingual models are identical. The models attend the screen, place the content in the imaginal buffer, and subsequently clear the buffer. Next, they attempt to retrieve the instructions from memory. This cycle is guided by ACT-R's temporal buffer (N. Taatgen, Van Rijn, & Anderson, 2008). The temporal buffer starts to accumulate *ticks* after the imaginal buffer has been cleared completely. The ticks act as ”counting points”, but each subsequent tick is slightly longer than the previous one. Early investigations showed

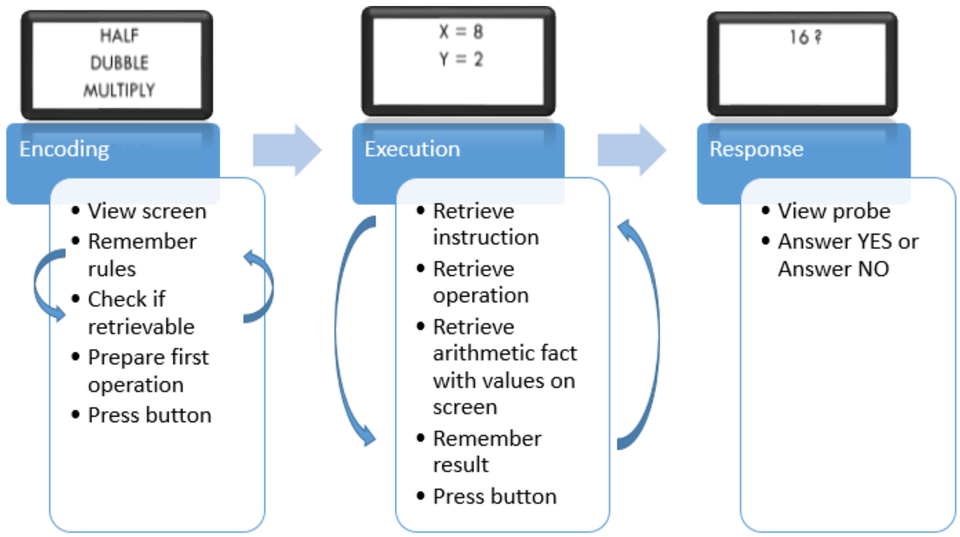

Figure 2: Overview of production flow through a trial. A set of three instructions (here: HALF DOUBLE MULTIPLY) is presented in the encoding phase. The models attempts to retrieve the instructions from declarative memory. If this is unsuccessful, the models take a new look and attempts again. Once the retrieval is successful, the instructions are copied in the imaginal buffer and first operation is prepared. The models press a button to continue. This process is identical in both the bilingual and monolingual model. During the execution phase, the models retrieve the instruction from the imaginal buffer, retrieve the appropriate operation (such as HALF = /2) from declarative memory, and retrieve the arithmetic fact (8/2 = 4) from declarative memory. The result of the operation is stored in the imaginal buffer and the models continue with the next operation. These steps (retrieve instruction, operation, fact, and storage in buffer) are separate in the monolingual model, but have been merged in the bilingual model (see also Appendix A). Once all operation have been performed, the models press a button, and continue with the response phase. The final answer is compared with the answer in the imaginal buffer, and the model responses accordingly. This process repeats until all trials have been completed.

that seven ticks resulted in a stable retention of the RITL-instructions. The model will repeat the encoding process if it has accumulated more than seven ticks when the RITL-instructions have been retrieved. The repetition increases the activation of the instructions chunk in declarative memory, which increases retrieval speed and likelihood. If the model successfully retrieves the instructions within seven or fewer ticks, the operation of the first instruction is prepared while pressing a button to continue.

2.2.4 Execution phase

As buffers contain only one chunk at a time, the models needs to swap information between buffers to store instructions and execute them later on. The strategy of swapping information between buffers is the essential difference between the monolingual and bilingual models (see also Table 1). The bilingual model's strategy is more efficient: it already combining rule application and variable binding at once, while the monolingual model needs to learn this first.

The bottleneck in intermediate storage becomes most apparent in the execution phase. The models retrieve an operation and do the calculations in similar fashion. The bilingual model stores the results of the calculations in the same chunk as the instructions. Thus, the instructions and results remain active in working memory, while retrieving answer to arithmetic operations at the same time. The monolingual model does not have this ability, and needs to retrieve the instructions again after retrieving an arithmetic fact.

The model retrieves results of the arithmetic operation from the declarative memory (here, the retrieval buffer). Arithmetic facts and knowledge of operations are assumed to be well established in the declarative memory.

2.2.5 Response phase

The response phases are identical again. The models check the result-chunks for a result. If the results corresponds to the probe on the screen, the models answer 'yes' using the keyboard. If not, they answer 'no' using the keyboard.

Because the models have been designed to retrieve the right instruction set, they won't make mistakes in this phase. Note that an actual accuracy of 1.0 is unrealistic. Bilinguals made slightly more mistakes than monolinguals in the experiment (respectively 87% and 92%), but this difference was not significant.

2.2.6 Model parameters

Table 2 presents the range under which parameter effects were examined and their function. The parameter's range refers to the range of values over which a full

Table 2: Parameters investigated in the ACT-R models.

Name	Function	Range	Default
alpha α	Learning rate	0.0 - 0.5	0.2
ans s	Activation and utility noise	0.01 - 0.05	-
imaginal-delay	Processing requests to imaginal buffer	0.15 - 0.25	0.2
le f	Retrieval exponent factor	0.9 - 1.1	1.0
nu ν	Starting utility for a newly learned production	0.0 - 0.5	0.0

grid search was conducted to explore the model behavior and evaluate the model performance.

2.3 Simulations

The models were presented with 100 identical data sets of 60 trials. Two combinations of instructions (always INCREMENT DOUBLE DIVIDE and TRIPLE INCREMENT ADD) are practiced 10 times each in random order. Immediately after these 20 trials, 40 trials are presented. Half of these consisted of practiced instructions and half of these are new instructions. The input numbers, which are presented in the execution phase, differed each trial.

Model selection was based on two approaches. First, we selected the model with the ACT-R's default values, except for parameter *ans (S)*. This parameter controls the activation and utility noise of productions and declarative knowledge. The other model selection method was binning the data set into five data sets for easier analysis and selection the model base on all parameters. Table 2 gives an overview of the parameters used in the simulations. For each unique combination of *alpha* α*, ans s, imaginal-delay, le f*, and *nu* ν, the model was run 100 times.

We used two approaches to select the most appropriate model to the data: using the ACT-R default values for the parameters, or choosing parameters with the best fit within the range presented in Table 2. The best fit to the results of the behavioral RITL experiment was found by identifying the unique set of parameter values that minimized the sum of squared errors between the mean responses times in the experiment by Stocco and Prat and the response times predicted by the models (Root Mean Square Error). We found a different parameter set when maximizing the correlation with the behavioral experiment. Both sets are presented in the next section.

3 Results

The model could decently fit the data (see Figure 3), but most importantly, could predict the decrease in response time.

3.1 Model Fit to Experimental Data

3.1.1 Root Mean Square Error

The root mean square error (RMSE) is calculated by squaring the root of the difference between the mean models' response times and the mean behavioral reaction times for language and instruction familiarity. Then, the RMSEs of the encoding and execution phases are added, and the minimal value is taken as the value with the smallest deviation from the behavioral data. See Table 3 for the resulting response times of the best fitting model. Table 4 shows that the smallest error was about 376 ms when all parameters are considered, and 864 ms when using ACT-R's default values. The lowest deviation for a single encoding phase was 53 ms, and for a single execution phase 290 ms.

Table 3: RMSE response times.

Type	Language	Encoding	Execution
Novel	Bilingual	2735	3442
Practiced	Bilingual	1964	3399
Novel	Monolingual	2680	4171
Practiced	Monolingual	1673	3887

Table 4: ACT-R model parameter values with smallest RMSE to the behavioral data of Stocco and Prat (2014). The first row contains the parameters with the smallest added Root Mean Square Error values. The second row shows the ans value with the smallest added RMSE; all other parameters were set to the ACT-R default value.

Alpha	Ans	Imaginal-Delay	le	nu	Error enc + ex (ms)
0.1	0.03	0.15	0.9	0.0	376
0.2	0.01	0.2	1.0	0.0	864

3.1.2 Correlation

The correlation between the behavioral data and the models is determined by adding the Spearman's rho for the variables of each task phase. Thus, the range of this metric is -2 to 2. See Table 5 for the resulting response times of the best fitting model. Table 6 shows that the largest correlation coefficient was 1.363 when all parameters are considered, and 1.247 when using ACT-R's default values. Overall, the models correlated better to the encoding phase than to the execution phase of the behavioral experiment. The highest Spearman's rho value for a single encoding phase was 0.998, but for a single execution phase was 0.511.

Table 5: Correlation response times.

Type	Language	Encoding	Execution
Novel	Bilingual	1861	3359
Practiced	Bilingual	1801	3278
Novel	Monolingual	1685	4323
Practiced	Monolingual	1631	3878

Table 6: ACT-R model parameter values with strongest correlations to the behavioral data from Stocco and Prat (2014). The first row shows the parameters with the largest added correlation value of both phases. The second row shows the ans value with the strongest correlation; all other parameters were set to the ACT-R default value.

Alpha	Ans	Imaginal-Delay	le	nu	Correlation enc+ex
0.1	0.05	0.20	1.0	0.0	1.363
0.2	0.02	0.20	1.0	0.0	1.247

3.1.3 A Middle Way

The two fitting methods show an surprising contrast: fitting the model parameters to the RMSE replicates the results for the encoding phase, while fitting to the correlation with the behavioral results replicates the results of the execution phase. Thus, it might be best to just find a middle ground between these two approaches. This resulted in a model with the parameters from Table 7, which yields reactions times of Table 8. See Figure 3 for a comparison of the experimental results to the model results. Notice that albeit the behavioral pattern is reproduced, the model does not exhibit the exact response times of the experiment.

Table 7: ACT-R model parameters

Alpha	Ans	Imaginal-Delay	le	nu	RMSE (ms)	Correlation enc+ex
0.1	0.04	0.20	1.0	0.0	867	0.842

Table 8: Response times.

Type	Language	Encoding	Execution
Novel	Bilingual	1999	3323
Practiced	Bilingual	1785	3275
Novel	Monolingual	2067	4266
Practiced	Monolingual	1531	3827

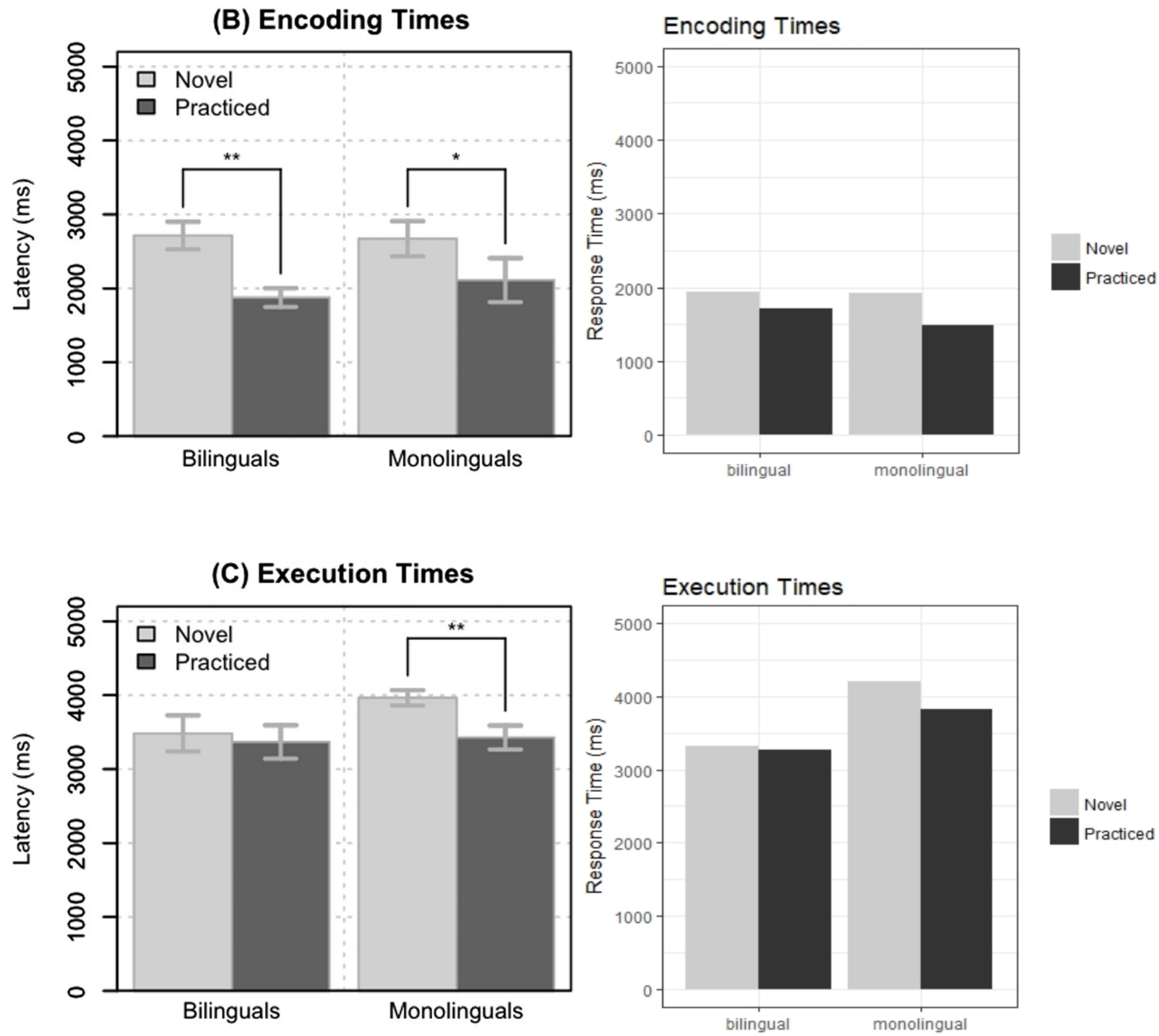

Figure 3: Comparison between the results from Stocco and Prat (2014) and model parameters that give a balanced result in both the encoding and execution phase. See Table 7 for parameter values.

3.2 Model Complexity

Besides finding the right parameters that produce the experimental outcomes, the model principles may stand regardless of parameter values. This can be examined through parameter space partitioning (Pitt, Kim, Navarro, & Myung, 2006). We expect that both models always produce a higher encoding RT in novel trials compared to practiced trials. The monolingual model should always produce a decrease in execution RT over the whole parameter space.

The parameter space is created by aggregating all possible parameter combinations. Then, we compare the differences in reaction time between novel and practiced instruction sets in each phase and variation of the model. Practiced trials have a lower encoding RT than novel trials for 46.5% of all parameter combinations in the bilingual model. In the monolingual model, on the other hand, practiced trials have a lower encoding RT in 68.7% of all parameter combinations. In the execution phase, the bilingual model produces lower RTs on practice trials in 74.8% of the trials. The monolingual model accurately predicts a difference between novel an practiced trials in the execution phase: 99.9% of all parameter combinations produce a lower execution RT on practiced trials.

3.3 Correspondence to Neural Correlates

ACT-R's modules are tied to specific brain areas. [I'D LIKE TO USE THE "BOLD SIGNAL" FROM THE DATA BUT I NEED TO DISCUSS HOW]

4 Discussion

The difference in task-switching behavior in bilinguals and monolinguals may result from differences in internal control, leading to a more efficient information exchange in bilinguals. Two ACT-R models were created to test this hypothesis; the monolingual model initially has more internal checks. More specifically, the model actively retrieves the next task in a separate production, but the bilingual model retrieves the next sub-tasks while storing information from the previous one. Over time, the monolingual model learns to act the same (see Table 1). The models could accurately reproduce the behavioral patterns observed in Stocco and Prat Stocco and Prat (Figure 3). However, the models could not predict the exact reaction times of the behavioral experiment. Thus, it seems that the model mechanism is accurate, but parameters or -more likely- production rules need to be refined to replicate the reaction times. We always refer to the "Middle Way Model" in subsequent sections, unless stated otherwise.

4.1 Encoding Phase

Parameter Space Partitioning (Pitt et al., 2006) shows that both models are unstable in the production of encoding RTs. The encoding times of practiced trials are lower in 47% of parameter variations for the monolingual model, and just 69% of trials presented to the bilingual model. On the other hand, both the Root Mean Square Error and correlation to the behavioral results are very favorable in this phase: the smallest RMSE in this phase was 53 ms and a Spearmann's rho of 0.998. No surprise: the variation was very small. For example, the overall Standard Error of the Mean of the "Middle Way Model" was a mere 12 ms. Thus, the models are capable of approaching the encoding times, but they also easily produce different patterns with minimal difference.

In other words, although the memorization mechanism seems to work, we cannot confirm that this is the way participants remembered the instructions. On the other hand, this mechanism mostly served to ensure that the instructions would retain in memory during the execution phase.

4.2 Execution phase

The execution response times are much more consistent: the difference in monolingual execution RTs is predicted in 99.9% of all parameter combinations. The bilingual model produces this pattern in 74.8% of all combinations, which may be connected to a slight task learning effects in bilinguals. Note that the performance of the monolingual model does not drop to the bilingual level, but decreases by 439 ms. Monolinguals were responded about 500 ms faster on practiced trials in the behavioral experiment. In short, the model mechanism can consistently predict a decrease in execution times and approach the response times we saw in Stocco and Prat (2014).

Production compilation and reduction of imaginal buffer checks can explain the neural results found by Stocco and Prat (2014). Bilinguals actively need to constrain themselves in processing novel instructions, perhaps to re-introduce internal checks and split up the procedure in separate steps - like monolinguals do when carrying out novel instructions. Thus, novel instructions may demand more from bilinguals than from monolinguals (see Figure 4, left). This also explains why the mean activation of bilinguals is so diverse when executing novel instructions: the graph may show individual differences in basal ganglia adaption. The right graph in Figure 4 shows that it's working: bilingual individuals with a higher mean beta value exhibit faster reaction times. The constant activation of the basal ganglia in monolinguals can be explained in a similar fashion in Figure 4: in both novel and practiced situations, monolinguals need to adapt in ways they haven't learned before. We observe this in the monolingual model behavior as well: new individual productions are created for individual instruction sets.

Note that the building strategy of the models influences the models' performance. The bilingual model is, essentially, the final product of the learning process of the monolingual model. It is therefor not surprising that these models behave similarly on practiced trials. It's safe to assume that the learning process bilinguals experience during a life time of language-switching is much more elaborate than the learning process of monolinguals when solving RITL trials. However, for the sake of simplicity, it is easier to assume that bilinguals and monolinguals only differ in their use of internal control. Stocco and Prat's participants were also sufficiently balanced. It therefore seems unlikely that participants substantially differed on aspects related to ACT-R's parameters, such as internal noise, starting utility for a new RITL procedure, or (non-linguistic) learning rate. I could not find any support for this idea in previous literature either.

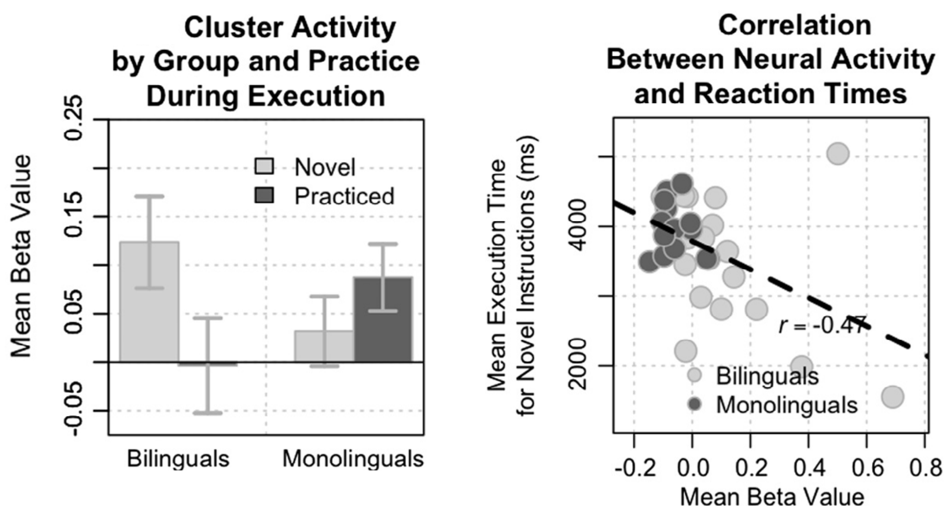

Figure 4: Left: Mean beta values (±SEM) during the Execution phase of the behavioral experiment in the left thalamus, left pallidus, and left putamen, divided by group (Bilinguals vs. Monolinguals) and practice (Novel vs. Practiced instructions). Right: Scatterplot illustrating the correlation between execution times and activation difference (Novel – Practiced trials) in the left putamen cluster of the basal ganglia. Retrieved from Stocco & Prat (2014).

4.3 The Role of the Basal Ganglia in Language-Related Differences in Task-Switching

Language-related differences in task-switching have been found in the prefrontal cortex, the anterior cingulate cortex, and the basal ganglia (Stocco & Prat, 2014; Abutalebi & Green, 2016; Seo et al., 2018). According to the conditional routing model, the basal ganglia steer information towards the prefrontal cortex if the conditions change. For bilinguals, this means that they need to choose between multiple rules for different languages, which puts additional strain of the basal ganglia circuit. The basal ganglia could be trained better to switch between sets of rules, and this skill may be ”flowing over” into non-linguistic tasks as well. This idea is further supported by the finding that the basal ganglia keep track of the target language during a bilingual RITL paradigm (Seo et al., 2018).

In the ACT-R architecture, the basal ganglia are related to the production module, which coordinates the interaction between other modules (Anderson et al., 2008, 2004) and are important in procedural learning. The prefrontal areas are associated with to the retrieval module, and the anterior cingulate with the goal module. The BOLD signal of the procedural module in our monolingual and bilingual models could show a patterns similar to Figure 4. They could also, however, show more activity during novel trials in the monolingual model; a pattern more similar to Figure 3.

[I NEED THE BOLD DATA FOR THIS SECTION]

4.4 Further Investigations

4.4.1 Accuracy

Although the bilingual group in Stocco and Prat was faster in the processing of novel trials, they made slightly more mistakes. This effect was not significant, but we may explain this with our model as well: they can be attributed to the removal of imaginal buffer checks resulting in the inability of retrieving the right set of instructions. Earlier versions of the models reflected this tendency, but were ultimately discarded, as accuracy is not the main focus of this investigation. However, other studies found this effect as well (Prior & MacWhinney, 2010), but the effect is not significant either. The models can be further adapted and used to investigate this marginal difference in accuracy in tasks in the future.

4.4.2 Applying model principles to other tasks

The monolingual model could reproduce the drop in response times when internal checks are removed and production compilation (N. A. Taatgen & Lee, 2003) was allowed. This opens doors for other task-switching paradigms, especially ones that have been modeled in ACT-R already, such as a traffic control task (N. A. Taatgen & Lee, 2003), or a multitasking paradigm that challenges ACT-R's information bottleneck (Borst, Taatgen, & Van Rijn, 2010). Borst et al. (2010) let participants alternate between typing a 10-letter word and subtracting a number. Earlier studies showed that people can only keep one intermediate task representation, or *problem state*, at the time, creating a task-switching bottleneck. In practice, that meant, for example, that participants had difficulty switching midway between typing the remaining letters of an obscured word. The behavioral results reflected this bottleneck. The task has also been modeled successfully using ACT-R. The authors later collected neuroimaging data during the task, and compared those to the "BOLD response" the model produces (Borst, Taatgen, Stocco, & Van Rijn, 2010; Borst & Anderson, 2015). The left SMA, which is associated to language switching (Abutalebi & Green, 2007, 2016), showed more activation in the hard conditions of the tasks. The prefrontal cortex, related to ACT-R's declarative module, showed an effect of task difficulty, but the expected interaction effect as a result of encoding problem states was not present. The authors suggested that this region's contribution to the processing problem states was perhaps too weak to impact the BOLD signal, or the retrieval of intermediate problem states is controlled by a different region.

The task has some overlap with the RITL paradigm: intermediate steps need to be stored and retrieved. Furthermore, the neuroimaging result of this task and the RITL paradigm can be compared, and may give insight in task-switching and the influence of the bilingual experience. If a bilingual and monolingual participant pool would perform the task, we may infer that bilinguals are more used to switching between information, and thus perform better in both difficult conditions of the task. Moreover, we may infer that bilinguals are initially faster than monolinguals in continuing with and finishing the task, as no queries of the imaginal buffer or declarative buffer, where problem state information is stored. However, monolinguals may catch up later. Again, this effect might be the result of reducing internal control in exchange for flexibility.

References

Abutalebi, J., & Green, D. (2007). Bilingual language production: The neurocognition of language representation and control. *Journal of neurolinguistics*, *20*(3), 242–275.

Abutalebi, J., & Green, D. W. (2016). Neuroimaging of language control in bilinguals: neural adaptation and reserve. *Bilingualism: Language and cognition*, *19*(4), 689–698.

Anderson, J. R. (2009). *How can the human mind occur in the physical universe?* Oxford University Press.

Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological review*, *111*(4), 1036.

Anderson, J. R., Fincham, J. M., Qin, Y., & Stocco, A. (2008). A central circuit of the mind. *Trends in cognitive sciences*, *12*(4), 136–143.

Bialystok, E. (2009). Bilingualism: The good, the bad, and the indifferent. *Bilingualism: Language and cognition*, *12*(1), 3–11.

Bialystok, E., Craik, F. I., Klein, R., & Viswanathan, M. (2004). Bilingualism, aging, and cognitive control: evidence from the simon task. *Psychology and aging*, *19*(2), 290.

Borst, J. P., & Anderson, J. R. (2015). Using the act-r cognitive architecture in combination with fmri data. In *An introduction to model-based cognitive neuroscience* (pp. 339–352). Springer.

Borst, J. P., Taatgen, N. A., Stocco, A., & Van Rijn, H. (2010). The neural correlates of problem states: Testing fmri predictions of a computational model of multitasking. *PLoS One*, *5*(9), e12966.

Borst, J. P., Taatgen, N. A., & Van Rijn, H. (2010). The problem state: A cognitive bottleneck in multitasking. *Journal of Experimental Psychology: Learning, memory, and cognition*, *36*(2), 363.

Carlson, S. M., & Meltzoff, A. N. (2008). Bilingual experience and executive functioning in young children. *Developmental science*, *11*(2), 282–298.

Cole, M. W., Laurent, P., & Stocco, A. (2013). Rapid instructed task learning: A new window into the human brain's unique capacity for flexible cognitive control. *Cognitive, Affective, & Behavioral Neuroscience*, *13*(1), 1–22.

Crinion, J., Turner, R., Grogan, A., Hanakawa, T., Noppeney, U., Devlin, J. T., … others (2006). Language control in the bilingual brain. *Science*, *312*(5779), 1537–1540.

Eurostat. (2018, 04 23). *Number of foreign languages known (self-reported) by sex.* (data retrieved from `http://appsso.eurostat.ec.europa.eu/nui/submitViewTableAction.do`)

Green, D. W., & Abutalebi, J. (2013). Language control in bilinguals: The adaptive control hypothesis. *Journal of Cognitive Psychology*, *25*(5), 515–530.

Luk, G., Green, D. W., Abutalebi, J., & Grady, C. (2012). Cognitive control for language switching in bilinguals: A quantitative meta-analysis of functional neuroimaging studies. *Language and cognitive processes*, *27*(10), 1479–1488.

MacWhinney, B. (1997). Second language acquisition and the competition model. *Tutorials in bilingualism: Psycholinguistic perspectives*, 113–142.

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.

Pitt, M. A., Kim, W., Navarro, D. J., & Myung, J. I. (2006). Global model analysis by parameter space partitioning. *Psychological Review*, *113*(1), 57.

Prior, A., & MacWhinney, B. (2010). A bilingual advantage in task switching. *Bilingualism: Language and cognition*, *13*(2), 253–262.

Seo, R., Stocco, A., & Prat, C. S. (2018). The bilingual language network: Differential involvement of anterior cingulate, basal ganglia and prefrontal cortex in preparation, monitoring, and execution. *Neuroimage*, *174*, 44–56.

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.

Stocco, A., Lebiere, C., O'Reilly, R. C., & Anderson, J. R. (2012). Distinct contributions of the caudate nucleus, rostral prefrontal cortex, and parietal cortex to the execution of instructed tasks. *Cognitive, affective, & behavioral neuroscience*, *12*(4), 611–628.

Stocco, A., & Prat, C. S. (2014). Bilingualism trains specific brain circuits involved in flexible rule selection and application. *Brain and language*, *137*, 50–61.

Stocco, A., Yamasaki, B., Natalenko, R., & Prat, C. S. (2014). Bilingual brain training: A neurobiological framework of how bilingual experience improves executive function. *International Journal of Bilingualism*, *18*(1), 67–92.

Taatgen, N., Van Rijn, H., & Anderson, J. (2008). Time perception: Beyond simple interval estimation. *Department of Psychology*, 60.

Taatgen, N. A., & Lee, F. J. (2003). Production compilation: A simple mechanism to model complex skill acquisition. *Human Factors*, *45*(1), 61–76.

United States Census Bureau. (2013, 08). *Language use in the united states: 2011.* (data retrieved from `https://www.census.gov/prod/2013pubs/acs-22.pdf`)

A Execution productions in monolingual and bilingual models

Monolingual

```
(p update-scratchpad-x
        =goal>
                isa phase
                step execution-x
        =retrieval>
                isa arithmetic-fact
                result =ans
        =imaginal>
                isa ritl-result
==>
        *imaginal>
                isa ritl-result
                x =ans
        =goal>
                isa phase
                step retrieve-task-y
)

(p retrieve-task-y
        ?imaginal>
        state free
        ?retrieval>
                state free
        =imaginal>
                isa ritl-result
                task1 =first
                task2 =second
                task3 =third
        =goal>
                isa phase
                step retrieve-task-y
==>
        =imaginal>
        =goal>
                isa phase
                step execution-y
        +retrieval>
                isa ritl-task
                kind ritl-task
                task1 =first
                task2 =second
                task3 =third
)

(p calculate-y
        =goal>
                isa phase
        step execution-y
        =imaginal>
                isa  ritl-result
                y nil
        =visual>
                isa  ritl-inputs
                y  =y
        =retrieval>
                isa  ritl-task
                task2 =second
==>
        =goal>
        =visual>
        *imaginal>
                isa ritl-result
                task =second
        +retrieval>
                isa operation
                task =second
                type unary
)
```

Bilingual

```
(p update-scratchpad-x-start-y
        ?retrieval>
                state free
        =goal>
                isa phase
                step execution-x
        =retrieval>
                isa arithmetic-fact
                result =ans
        =imaginal>
                isa  ritl-task
                task2 =second
==>
        *imaginal>
                isa ritl-result
                x  =ans
        +retrieval>
                isa  operation
                task =second
                type unary
        =goal>
                isa phase
                step execution-y
)
```

Monolingual

```
(p update-scratchpad-y
        =goal>
                isa phase
                step update-scratchpad-y
        =retrieval>
                isa arithmetic-fact
                result =ans
        =imaginal>
                isa ritl-result
                y nil
                task1 =first
                task2 =second
                task3 =third
==>
        *imaginal>
                isa ritl-result
                y =ans
        =goal>
                isa phase
                step retrieve-task-binary
)
```

Monolingual

```
(p retrieve-task-binary
        ?imaginal>
                state free
        =imaginal>
                isa ritl-result
                task1 =first
                task2 =second
                task3 =third
        =goal>
                isa phase
                step retrieve-task-binary
        ?retrieval>
                state free
==>
        =imaginal>
        +retrieval>
                isa ritl-task
                kind ritl-task
                task1 =first
                task2 =second
                task3 =third
        =goal>
                isa phase
                step execution-binary
)
```

Bilingual

```
(p update-scratchpad-y-start-binary
        ?retrieval>
                state free
        =goal>
                isa phase
                step update-scratchpad-y
        =retrieval>
                isa arithmetic-fact
                result =ans
        =imaginal>
                isa ritl-result
                task3 =third
                y nil
==>
        *imaginal>
                isa ritl-result
                y =ans
        +goal>
                isa phase
                step execution-binary
        +retrieval>
                isa operation
                task =third
                type binary
)
```

Monolingual

```
(p calculate-binary
        =goal>
                isa phase
                step execution-binary
        =imaginal>
                isa ritl-result
                result nil
        =retrieval>
                isa ritl-task
                kind ritl-task
                task3 =third
==>
        *imaginal>
                isa ritl-result
                task =third
        =goal>
        +retrieval>
                isa operation
                task =third
                type binary
)
```

B Afterword

I would like to thank dr. Jelmer Borst and dr. Andrea Stocco for their guidance during this project. Special thanks go to all the people at the University of Washington's Cognition & Cortical Dynamics Laboratory for their warm welcome and support during my stay.