DOCTORAL THESIS

Facilitating the Use of and Extending the RT-MPT Model Class

A thesis submitted in fulfillment of the requirements for the degree of Doctor of Natural Sciences in the

Faculty of Economics and Behavioral Sciences

of the

Albert Ludwig University of Freiburg i. Br.

submitted by

Raphael HARTMANN

Zurich, CH

supervised by

Prof. Dr. Andrea KIESEL

and

Prof. Dr. Karl Christoph KLAUER

Winter Term 2020/2021

Dean of the Economics and Behavioral Science Faculty:
Prof. Dr. Olaf N. Rank

Thesis Reviewers:
Prof. Dr. Andrea Kiesel
Prof. Dr. Karl Christoph Klauer

Thesis Defense:
January 18, 2021

Defense Committee:
Prof. Dr. Karl Christoph Klauer
Prof. Dr. Andrea Kiesel
Prof. Dr. Daniel Heck

Declaration on Oath

Declaration on Oath in accordance with section 7 paragraph 1 clause 3 no. 9 of the Doctoral Regulation of Albert Ludwigs University for the Economics and Behavioral Science Faculty.

1. The thesis submitted on the topic *Facilitating the Use of and Extending the RT-MPT Model Class* is my own achievement.
2. I have only used the sources and resources specified and have not had any third party help. In particular I have identified where I have adopted the literal or conceptual work of others.
3. I have presented thesis or a part thereof as yet to no other university in Germany nor abroad as subject of an examination or qualification achievement.
4. I confirm that the above declaration is correct.
5. I understand the significance of this declaration on oath and I am aware of the criminally punishable consequences of making a false or incomplete declaration on oath.

I, Raphael HARTMANN, swear on oath that to the best of my knowledge that I have declared the whole truth and have not concealed anything.

Place and date | Signature

Acknowledgements

I would like to thank the numerous people who helped me along the way of this dissertation. It would not have been possible without you and I am truly grateful for each and every one of you.

First and foremost, I want to thank my two advisors, Andrea Kiesel and Christoph Klauer. I could not have asked for better advisors. I believe it was the perfect combination of advisors. You taught me so much.

I want to thank Andrea Kiesel for making me feel welcome and for the many times you gave me advice. I want to thank you for your kindness and generosity. I don't take it for granted that you gave me your open slot in SMiP and therefore the opportunity to do my PhD in Freiburg.

I thank Christoph Klauer for all his great efforts in helping me, especially when I got stuck and did not know how to proceed. I am thankful for your kindness, your generosity, and patience with me. You gave me the opportunity to work on something I really enjoyed, you always took the time to explain complicated methods, and to think about a solution to my problems. I learned a lot from you and for that I am very thankful.

I thank Victor Mittelstädt for being the best swimming companion. You taught me the valuable lesson of making breaks and find a work-live balance. It was nice having you as an office mate.

I also thank Niels Kukken with whom I spent the best times in Mannheim, Heidelberg, and Tübingen. I was so glad to have you there. It was nice to discuss the topics of the workshops and to have someone I can talk to also about personal stuff.

Another thanks goes to Constantin Meyer-Grant with whom I could discuss my work and progress. I am glad that you are here and that I can also discuss more mathematical and technical stuff with you. It is also nice to finally have someone with whom I can go bouldering.

I want to thank one special person, who does not want to be named here, for making my life in Freiburg much more pleasant. I don't know what I would have done without you all this time. I thank you for the great time with you and your help with stuff outside academia.

I also thank the people of my two labs (Cognition, Action, and Sustainability as well as Social Psychology and Methodology). I thank André and Annelie for taking the time to listen to my problems. Especially at the beginning of my PhD I was so glad that I could talk to you. I also thank Anne and Manu for helping me with practical stuff like teaching. I tend to overthink stuff and you could provide me with all the information I needed. I also thank Rainer for his many help and his patience with me. Another thanks goes to Danielle who made the change to Freiburg easier because I could still speak Swiss German from time to time. I also thank the Secretaries, Angelika and

Sandra, for their help with the administrative stuff that would have caused me many headaches without you.

I thank SMiP for giving me this great opportunity. I value the workshops and courses you provided us with. I thank Anke Söllner and Annette Förster for their help, whether with hotel booking or answering questions about administrative stuff. I also thank the many PhD candidates and PIs with whom I had conversations. A special thanks goes to Edgar Erdfelder and Daniel Heck for their great help.

You cannot choose your family, but I would have chosen mine anyways. I am thankful for my two brothers. It is so good to know that I can always rely on you. I also thank my mother for always being there for me. I cannot express how thankful I am for having you. You are my save haven. I also thank my father for all his support he gave me. I like to think, you would have been proud of me. This dissertation is for you, mum and dad.

Summary

The multinomial processing tree (MPT) model class is well known in cognitive psychology. With models of this class we try to explain the underlying cognitive mechanisms of decision tasks. These mechanisms include cognitive processes and describe how they interact with one another. From the frequencies of the responses it is possible to estimate the probabilities of the process outcomes.

The Response-time extended MPT (RT-MPT; Klauer & Kellen, 2018) model class extends the MPT model class by using response times in addition to the response frequencies. It allows for the estimation of process completion times – the time it takes for a process to complete with a given outcome – and the encoding and motor-execution times – also known as non-decision time. Each process has two or more outcomes and each process outcome in turn has an assigned probability and a respective completion time. The first implementation of RT-MPT was in *C++* which made it rather inaccessible for potential users, except the developers, and contained a proprietary library. It was therefore neither easy to use nor free to get.

In this thesis, I facilitate the use of the RT-MPT model class. In three articles, I and my coauthors develop a software package that is easy to use, free, and open source, extend the software package to allow for fitting even more RT-MPT models, validate the algorithm of the software package, and lay a basis for efficiently modeling RT-MPT models that assume Wiener diffusion processes. In our first project we integrate the RT-MPT model class in the statistical programming language *R* by developing a so called *R* package. In this project we also validate the resulting software package and test whether it still produces the same results as in the original paper by Klauer and Kellen (2018). In our second project, we extend the model class – or rather its implementation – to allow for fitting RT-MPT models with repeating processes on a path. Due to some properties of the old likelihood function this was not possible before. We validate the new algorithm (with a modified likelihood function) to check whether the new algorithm is still working properly. Finally, in the last project, we derive partial derivatives of the first-passage time density and cumulative distribution function of the Wiener diffusion model. As Klauer and Kellen (2018) suggest, the Wiener diffusion model might be a promising framework for modeling cognitive processes. This would be a competing alternative to modeling process times with exponential distributions as in Klauer and Kellen (2018). Partial derivatives provide information that is needed for efficient parameter estimation procedures.

Contents

List of Figures

List of Abbreviations

2HTM	**Two-H**igh **T**hreshold **M**odel
CDF	**C**umulative **D**istribution **F**unction
GPT	**G**eneralized **P**rocessing **T**ree
MCMC	**M**arkov **C**hain **M**onte **C**arlo
MPT	**M**ultinomial **P**rocessing **T**ree
PDF	**P**robability **D**ensity **F**unction
RT	**R**esponse **T**imes
RT-MPT	**R**esponse **T**ime extended **M**ultinomial **P**rocessing **T**ree
RV	**R**andom **V**ariable

Dedicated to my parents

Chapter 1

Introduction

1.1 Multinomial Processing Tree Models

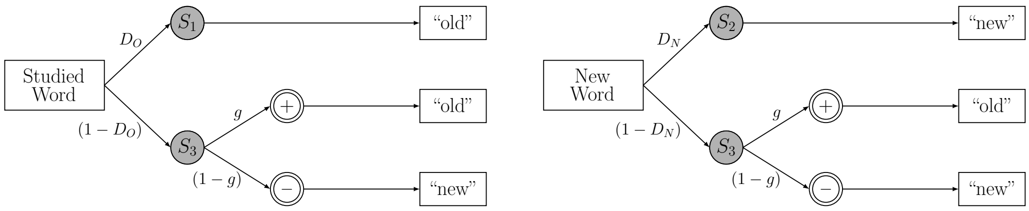
In cognitive psychology we are interested in cognitive mechanisms and processes underlying the completion of tasks. The class of multinomial processing tree (MPT) models or multinomial models as introduced by Riefer and Batchelder (1988) is a class of models with which we can investigate such mechanisms and processes by using frequency data. It is assumed that for a person to complete a task, e.g., a *word recognition* task, there are certain necessary cognitive processes involved. Each process can, with certain probabilities, lead to different outcomes which in turn might trigger other processes. As Riefer and Batchelder (1988) state, these cognitive processes are not directly observable. When employing MPT models we try to capture the relevant cognitive processes and the cognitive mechanism underlying these processes.

To get a better understanding of what is meant by cognitive mechanisms and processes we will take a look at an example. In a simple word recognition task participants try to recall words from a studied list and try to distinguish them from new words. The task is simple; in each trial, they have to state whether the presented word is *old* (i.e., studied), or *new*. Even this seemingly simple task requires multiple cognitive processes and a mechanism that defines how these processes interact with each other. The cognitive processes involved might be a process for detecting studied words, a process for detecting new words, and a guessing process. MPT models trying to capture the mechanism behind such tasks are called threshold models (Krantz, 1969; Luce, 1963; Snodgrass & Corwin, 1988; Swets, 1961). For example, the *two-high threshold* model (2HTM; Snodgrass & Corwin, 1988) assumes that words are either detected by the corresponding detection process, or will be guessed. Therefore, detection can either lead to a success or a failure, and guessing can either lead to a tendency to respond with "old" or "new". Only if detection fails a wrong answer might be produced by guessing, i.e., detection dominates guessing. The sequence with which the processes occur (detection-first or guessing-first, see Figures 1.1 and 1.2) can also be considered as part of the cognitive mechanism but cannot be distinguished within the MPT framework. The two representations – or rather variants – of the model are mathematically equivalent. This can be seen by comparing the probabilities for a miss in both representations; in the Detect-Guess variant (i.e., detection-first) this is $(1 - D_O) \times (1 - g)$ and in the

Default-Interventionist variant (i.e., guessing-first) this is $(1-g)\times(1-D_o)$, which is the same multiplication, but with a reversed order. The probabilities of a false alarm for the respective variants are $(1-D_N)\times g$ and $g\times(1-D_N)$, respectively.

FIGURE 1.1

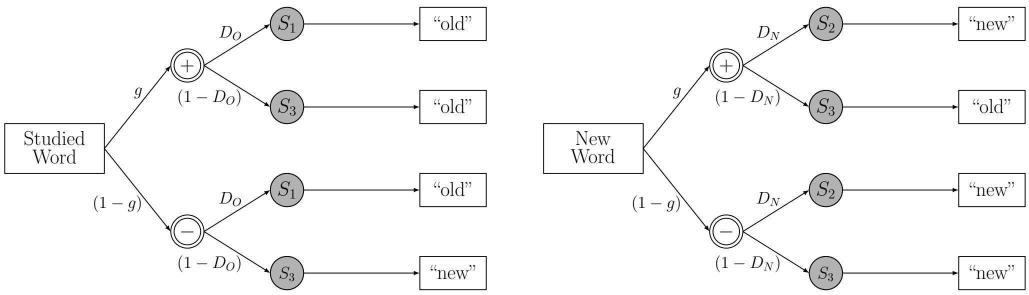
2HTM With Detection First a.k.a. Detect-Guess



Note. This figure is a modified version from Hartmann et al. (2020). Instead of using the general notation for probabilities θ_p abbreviations of the process names are used: Parameter g denotes the process-probability for guessing "old", D_O for detecting a studied word as "old", and D_N for detecting a new word as "new". The circles indicate latent states; + and − denote biases/tendencies towards the responses "old" and "new", respectively, S_1 and S_2 states of certainty for old and new words, respectively, and S_3 a state of uncertainty. The rectangles indicate observable categories; On the left of each tree the trial categories and on the right the response categories.

FIGURE 1.2

2HTM With Guessing First a.k.a. Default-Interventionist



Note. This figure is a modified version from Hartmann et al. (2020). See Figure 1.1 for more details.

The interpretation of MPT models is quite intuitive. With only a few parameters, representing the process probabilities (Riefer & Batchelder, 1988), and some if-then structures we can describe how the underlying processes interact with each other to form an answer, i.e., describe the cognitive mechanism we assume. Its simplicity has made the model class of MPTs very prominent in cognitive psychology and beyond (for reviews see Batchelder & Riefer, 1999; Erdfelder et al., 2009; Hütter & Klauer, 2016).

Despite its use and simplicity the MPT model class has some limitations. First, latency information is not used to further improve inference. Even though for each trial the reaction times (RTs) are often measured, this information is not integrated in MPTs. Second, as already briefly mentioned above, the sequential order of the processes cannot be tested. Whether the 2HTM starts with an initial guessing – or rather bias – (see Figure 1.2) or directly with a detection attempt (see Figure 1.1), makes no difference mathematically. Third, often MPT models are not identified (Klauer & Kellen, 2018).

This, for example, happens when there are more parameters in the model than non-redundant frequency categories (e.g., false alarms in new word trials). This is also the case in our example; the 2HTM represented in 1.1 has three parameters but only two non-redundant frequency categories (misses in old word trials and false alarms in new word trials). Unfortunately, even if there are as many or more non-redundant frequency categories than parameters, the model might still not necessarily be identified, i.e., there are more conditions that must be met, such that an MPT model is identified. For more details about identifiability see Schmittmann et al. (2010).

All of these mentioned limitations can be tackled by integrating RTs into MPT models. In the following section I will therefore briefly describe some approaches to do so and illustrate their respective advantages and disadvantages.

1.2 RT-Extensions of the MPT Model Class

As early as 150 years ago, Donders (1868/1969) tried to measure process times with his *method of subtraction*. Even though this procedure was restricted to measure the completion time of only one cognitive process it still was an inspiration for other approaches.

One such approach was proposed by Hu (2001) who generalized an earlier approach by Link (1982) to integrate RTs in MPT models. In order to do that he used mean RTs per category and mixture distributions to disentangle the mean process-completion times – henceforth process times. Because of identifiability reasons this procedure was restricted to setting the process times for all possible outcomes of a process to be equal. In addition, encoding and motor-execution times – henceforth motor times – are not considered. Nevertheless, this was a first large step into modeling latencies alongside frequencies.

Another, more recent approach similar to the one by Hu (2001) was provided by Heck and Erdfelder (2016). They developed a model class using a non-distributional method to integrate RTs. The observed RTs (of each person separately) are split into bins which are used as additional categories. This means that at the terminal nodes of traditional MPT models an additional process is attached leading to the different RT bins. With this procedure it is possible to estimate "relative" process times. In our simple word recognition example with the 2HTM this means one could test whether answering correctly to a studied word is faster with detection than with guessing. One downside of this approach is that it is usually not identified if the underlying traditional MPT is not identified.

A third noticeable approach is the generalized processing tree (GPT) model class by Heck et al. (2018). In GPT models not only RTs but also other continuous random variables (RVs) can be integrated. Each of these continuous RVs are modeled jointly with the frequencies. For each possible path a distribution can be assigned which the RTs (or other continuous RVs) are assumed to follow. Heck et al. (2018) use a mixture distribution with the branch probabilities as weights to estimate properties of the branches. This approach is still not able to give separate estimates for the process times and motor times. Nevertheless it can give a rough estimate of the decision and non-decision times by using, for example, the ex-Gauss distribution and taking the mean parameter of the Gauss component as an estimate for the non-decision time and the inverse of the rate parameter of the exponential component as a decision time. However, even if the underlying traditional MPT model is not identified, the GPT model might still be.

The final approach by Klauer and Kellen (2018), called RT-extended MPT (RT-MPT), also models RTs and frequencies jointly but in a slightly different fashion. The completion time of each process outcome is assumed to be exponentially distributed and the motor times nomally distributed with a truncation from below at zero. The RT for each path is therefore assumed to be a sum of one or more exponential RVs and a RV following a truncated normal distribution. In a sense, this model class can be seen as a special case of GPTs. Nevertheless, there are some fundamental differences, which will be apparent in the next section. One important feature of RT-MPTs by Klauer and Kellen (2018) is that one can test the sequential order of the processes. Therefore one can test, for example, which process comes first in the 2HTM, detection or guessing (see Figures 1.1 and 1.2). In addition, many MPT models can become identified by using RT-MPTs.

In the following section the model class of RT-MPT by Klauer and Kellen (2018) will be described in more detail. This model class builds the basis for this dissertation.

1.3 RT-MPT Model Class

The approach by Klauer and Kellen (2018) uses a hierarchical Bayesian structure with groups and persons within groups. It assumes two additional latent variables for each person s compared to the MPT model class, namely the process times τ_s, and the motor times δ_s. Each process p can have two potential outcomes o and there can be a different motor time for each response r. As already mentioned above this leads to

$$\begin{aligned} \tau_{p,o,s} &\sim Exp\left(\lambda_{p,o,s}\right) \\ \delta_{r,s} &\sim TN_{\geq 0}\left(\gamma_{r,s}, \sigma_s\right), \end{aligned} \tag{1.1}$$

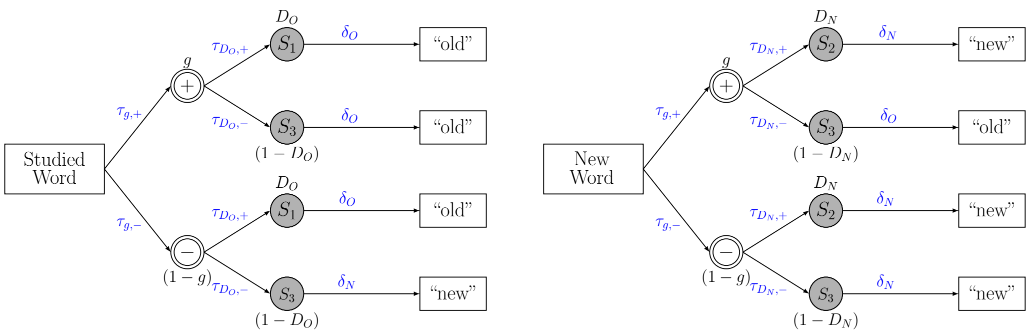
where Exp denotes the exponential distribution and $TN_{\geq 0}$ the normal distribution truncated from below at zero. There are four parameter vectors of interest – the process-outcome probabilities θ which is already used in the MPT model class, the rates λ of the process-time distribution, as well as the mean parameter γ and standard deviation σ of the motor-time distribution.

Let us come back to our 2HTM example and take a look at the variant with guessing first. For all branches leading to the response category “old” we might assume that person s has a motor time $\delta_{O,s}$ and for all branches leading to the response category “new” motor time $\delta_{N,s}$. The time for person s to complete the detection process of an old word with a success is denoted by $\tau_{D_O,+,s}$ and with a failure it is denoted by $\tau_{D_O,-,s}$.

Similar notation is used for the detection process of new words with $\tau_{D_N,+,s}$ and $\tau_{D_N,-,s}$ for success and failure, respectively. For the guessing process $\tau_{g,+,s}$ and $\tau_{g,-,s}$ denote the process times for a guessing in favor for the response "old" and "new", respectively. See Figure 1.3 for a graphical representation of the RT-extended 2HTM.

FIGURE 1.3

RT-2HTM With Guessing First



Note. This figure is a modified version from Hartmann et al. (2020). See Figure 1.1 for more details about the structure and the process-probability parameters. In blue color the different types of time parameters are depicted. The τ's represent process times with the abbreviated process names (detection old/new and guessing) and their respective outcomes as subscript and the δ's represent the motor times with the abbreviated response names ("old" and "new") as subscripts. The subscripts for a person s are omitted.

For the joint density of the response categories (c) and RT (t) in Klauer and Kellen (2018) I need to introduce some additional formulae; let us start with the probability of path B, $\mathbf{P}(B)$. It is just the product of all process-outcome probabilities on a path. In our example the path probability for responding "old" to an old word with uncertainty is $\theta_{g,+,s} \times \theta_{D_O,-,s}$, or in the simple notation without person subscripts $g \times (1 - D_O)$, where g denotes the probability to guess "old" and D_O the probability to detect an old item.

Next we need the density of the RT, given a path B, $f(t|B)$. In Klauer and Kellen (2018) this density is defined by the convolution of the exponential distributions of the involved process times and the truncated normal distribution from the motor time on a path B. This convolution is a *modified ex-Gauss* distribution for paths consisting of only one process time. Modified because a *truncated* normal distribution is used compared to the known ex-Gauss distribution (e.g., Matzke & Wagenmakers, 2009). When more than one process time is involved, we call it *modified hypoex-Gauss* distribution (for more details about the modified hypoex-Gauss distribution see e.g., Hartmann et al., 2020). In our last example, it would be a modified hypoex-Gauss distribution arising from the convolution of three distributions, namely the distributions of $\tau_{g,+}$, $\tau_{D_O,-}$, and δ_O (omitting the subscript for person s).

The joint density of the RT and response category in Klauer and Kellen (2018) is then

$$f(c,t) = \sum_{B:\ B\ ends\ in\ c} f(t|B)\mathbf{P}(B). \tag{1.2}$$

This joint density and the conditional density of the RTs given a path will become important in Chapter 3 where it needs to be modified and in Chapter 4 where the conditional density has no closed form, meaning it can only be calculated by numerical integration.

Let us now turn to the most important prior choices of the parameters in the RT-MPT model class by Klauer and Kellen (2018), namely the ones for the process probabilities θ, process rates λ, and mean parameter of the motor times γ. The core feature is that for each person the parameters come from a group mean and a person-specific deviation from that mean. However, the parameters are first transformed, such that they are on the same scale $(-\infty,\infty)$. Klauer and Kellen (2018) define $\alpha = \Phi^{-1}(\theta)$ and $\beta = log(\lambda)$. The separation of group mean and person specific deviation is then given by:

$$\begin{aligned}\alpha_{p,s} &= \mu_p^{(\alpha)} + \alpha'_{p,s}\\ \beta_{p,o,s} &= \mu_{p,o}^{(\beta)} + \beta'_{p,o,s}\\ \gamma_{p,s} &= \mu_r^{(\gamma)} + \gamma'_{r,s},\end{aligned} \tag{1.3}$$

where the μ's denote the respective group mean, and $\alpha'_{p,s}$, $\beta'_{p,o,s}$, and $\gamma'_{r,s}$ denote the person specific deviations from the group means.

For each person the process-related parameters in Klauer and Kellen (2018) are assumed to correlate. This is implemented by using a multivariate normal prior with mean zero and variance-covariance matrix Σ for the vector (α'_s, β'_s). The parameter $\mu_p^{(\alpha)}$ follows a normal distribution with mean zero and ϵ times an identity matrix as variance-covariance matrix, where ϵ is a scaling parameter. Klauer and Kellen (2018) decided that for the process rates the group-level parameter is transformed back to the original scale, $exp(\mu_{p,o}^{(\beta)})$, and follows an independent gamma distribution with shape and rate parameters set to one and 0.1, leading to a mean process time of 10 ms and a variance of 100. The parameter vector γ'_s follows a multivariate normal distribution with mean zero and variance-covariance Γ, allowing the motor times to correlate for each person, and $\mu_r^{(\gamma)}$ follows a normal distribution with zero mean and ten times the identity matrix as a variance-covariance matrix.

There is one special feature of the RT-MPT model class by Klauer and Kellen (2018) I would like to outline here as well. For their inference they used a Markov chain Monte Carlo (MCMC) algorithm. In particular, they used a Metropolis-within-Gibbs sampler (e.g., Gelfand & Smith, 1990). This means, that whenever possible they use a Gibbs sampler and otherwise a Metropolis-Hastings sampler. The Gibbs sampler is a special case of the Metropolis-Hastings sampler that allows to sample from a multivariate distribution, $f(x_1, x_2, \ldots, x_d)$, without rejecting any sample, but requires the conditional distribution of one parameter given all other parameters, $f(x_i|\{x_j : j \neq i\})$, which does not always have a closed form. What can help is using a so-called *data augmentation* method (Albert & Chib, 1993; Tanner & Wong, 1987). Data augmentation is, according to Tanner and Wong (1987), a scheme of augmenting observed data to make analysis easier. He points out that in ideal cases one can sample from the posterior given the augmented data with ease. In Albert and Chib (1993) it is used in the context of probit regression; they use latent normal variables, Z_i, for each trial i to get the fully conditional posterior density of the regression parameter β, $f(\beta|y,Z)$, which is the same as for a normal linear model $Z = X\beta + \epsilon$. Using this conditional distribution, they developed a Gibbs sampler. Klauer and Kellen (2018) use a similar data-augmentation approach to develop a Gibbs sampler for the process probabilities – or rather the probit transformed probabilities α. Their procedure goes further to also make a Gibbs sampler for β (log-transformed process rate parameters) and γ (mean parameter of the motor times). This data augmentation will become important for Chapter 4.

1.4 Goals and Subgoals

Even though the RT-MPT model class was implemented very efficiently in the programming language *C++* it was not straight forward to fit an RT-MPT model to some data. Furthermore, it required at least some knowledge of *C++* and a proprietary *C++* library was used for optimizations and other algebraic routines.

The first goal of this dissertation was, therefore, the implementation of RT-MPT into a more widely used programming language, namely as a software package of the programming language *R*. There were several aspects that needed to be considered. First, the usability; it should be as usable as possible, preferably by employing similar conventions as used for other *R* packages for Bayesian inference and/or modeling MPT models. Second, it should be free and open source; no proprietary software (packages) should be included and the code should be made publicly accessible. Third, its validation; the underlying algorithm should be validated in order to guarantee accurate estimates. Beyond that, the speed of the software package was also an important aspect.

The next goal was the extension of the model class to enable it to work for as many (RT-)MPT models as possible, i.e., it should not be restricted to only a subgroup of (RT-)MPT models. For that an additional validation was necessary.

The last goal was to create the basis for modeling cognitive processes within the RT-MPT structure with a Wiener diffusion model using gradient information. The reason behind this was the relatively strong restricting assumption that the process times follow an exponential distribution.

Chapter 2

Implementation of RT-MPT in *R*

As already mentioned, in Hartmann et al. (2020)[1], we decided to implement the RT-MPT model class directly into *R* by developing a package that uses the already existing *C++* code written by Klauer and Kellen (2018). The *R* environment serves as a wrapper for the *C++* code, i.e., everything can be accessed from *R*, which in turn executes the *C++* program.

The communication between these two programming languages is established by a function written in *C*. This function takes *R* objects as pointers – meaning it takes the memory addresses of the objects as variables – and can then be used within *C*/*C++*. In that way, the objects must not be copied and can be directly used for any sort of calculation. Once the calculations within *C*/*C++* are finished, the function then returns a new *R* object to *R* with the desired results.

There were many subgoals for this project. The accessibility of the software package, its usability, and the validation of its algorithm. Before I will discuss these subgoals in the following sections, I will describe our first attempts to implement RT-MPT in *R* in the next section.

2.1 Attempts

Our first idea was to implement the RT-MPT model class into an existing Bayesian software package within the programming language *R* (R Core Team, 2020). This would have been the most straight forward way to make the model class accessible to potential users.

The first software package we used was `rstan` (Stan Development Team, 2018). Even though this relatively new software package was very promising, it had the problem that the likelihood function is required to be differentiable. Since this was not the case, it led to so-called *divergent transitions*, which in turn led to biased estimates.

[1]Raphael Hartmann is credited as the main author of this article. He wrote the *R* code for the *R* package `rtmpt`, adjusted some of the *C++* code to enable the communication between *R* and *C++*, performed the simulations and analyses and wrote the first draft of the manuscript. Lea Johannsen contributed with the idea and setting of the validation study, wrote a script for the graphical visualization thereof, and provided recommendations for improving the manuscript. Karl Christoph Klauer contributed by modifying the *C++* code, provided suggestions for the simulations studies, advised the analyses, and provided recommendations for structuring and improving the manuscript.

The second software package we used was `rjags` (Plummer, 2018). Even though we managed to implement the likelihood function as an additional JAGS *module* (for developing such modules see Wabersich & Vandekerckhove, 2013) there were problems with convergence even for very simple models.

Only after these attempts we decided to directly implement the model class into *R* using the already existing *C++* code.

2.2 Accessibility

The original *C++* source code contained a lot of routines from a proprietary library, since the computations require operations from linear algebra and also some optimizations. Such a library would have restricted the accessibility to people having a necessary licence. Therefore we decided to replace this library with a free and open source library, namely the GNU Scientific Library (GSL; Galassi et al., 2009).

The choice for using an *R* package as a wrapper over the *C++* code is similar. *R* is a free and open source programming language for statistical computation (R Core Team, 2020). In our opinion it is one of the most used statistical software in psychology. *R* packages provide an easy way to make functions accessible to many researchers. They can be easily installed an used.

2.3 Usability

The RT-MPT model class should be as easy to use as possible. This is another reason why the programming language *R* is ideal. It consists of only a few data types, namely `numeric` (integer as well as real numbers), `logical` (*true* and *false*), and `character` (strings). All other *R* objects (e.g., array, vectors, and lists) consist of these types of objects. All *R* objects can be used to define (new) functions. This rather simple structure makes it very easy to use *R*.

Broadly speaking, *R* packages are nothing more than a collection of *R* functions that can be installed and loaded with simple commands. Once a package is installed and loaded all functions within it can be used.

For these reasons we developed an *R* package called `rtmpt` (Hartmann et al., 2020) with some functionalities to enable other researchers to fit RT-MPT models to data. It consists of mainly five functions. One function transforms the data and one transforms the model file into the right format, such that they can be read within the *C++* program. Two functions allow for optional changes in the model specifications/restrictions. The main function fits the model to the data, which is provided by the other functions.

To make the use of the main function as easy as possible we used similar notations to other *R* packages using Bayesian inference like `rjags` (Plummer, 2018). For example, the number of chains denoted with `n.chains` or the number of iterations denoted with `n.iter`. For more details about the package, see Hartmann et al. (2020).

2.4 Algorithm Validation

Since we decided to implement the RT-MPT model class directly into *R* using the already existing *C*++ code, it was necessary to validate the underlying algorithm. Therefore, we wanted to answer whether the algorithm is suitable for the RT-MPT model class.

For Bayesian inference using MCMC methods, one way to validate an algorithm is by using *simulation-based calibration* (Talts et al., 2018). Similar to traditional recovery studies datasets are randomly sampled for each of the I repetitions. The difference is to chose a prior distribution from which the true values are sampled for each repetition, unlike in traditional recovery studies where only one set of true values is used. These true values are then used to simulate data. Talts et al. (2018) argue that by using the same priors for the Bayesian model and because of the *self-consistency* condition the data-averaged posterior distribution should equal the prior distribution, if the algorithm is valid for the specified model. This is tested by histograms of ranks. For each of the N repetitions and each parameter independent samples are drawn/selected from the posterior distribution and compared to the ground truth. If the $N+1$ possible ranks through all the repetitions differ significantly from a uniform distribution then we can say that the algorithm is not valid for the model and should not be used.

In contrast, according to Talts et al. (2018), traditional recovery studies do not allow for such conclusions. Even if the algorithm is not suitable for the inference the results for the recovery might still suggest everything being acceptable. The same holds true for the opposite. Even if the results suggest problems with the algorithm it might still be valid.

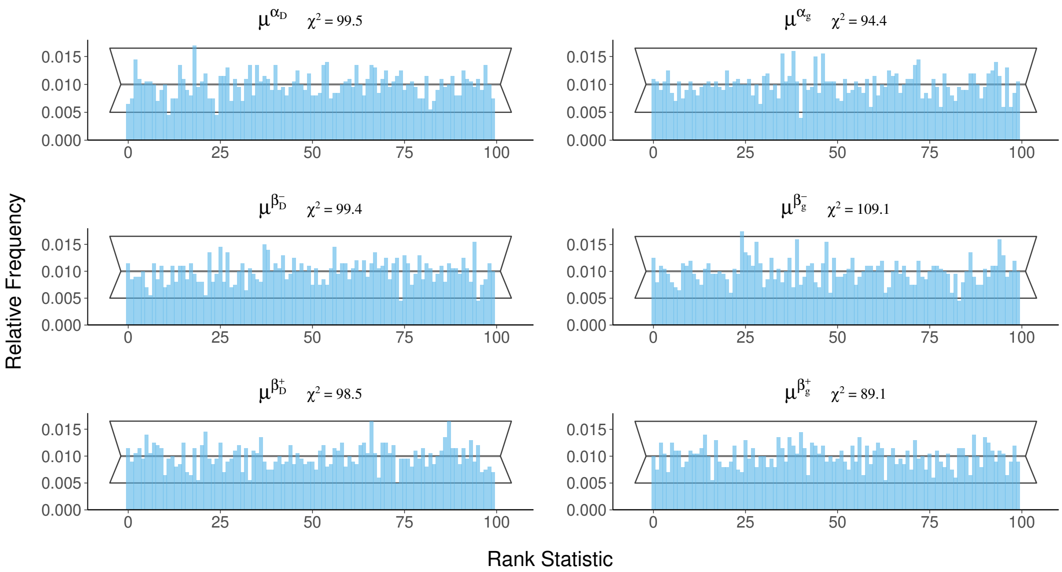
Figure 2.1 depicts the histograms of the rank statistics for the group-level process-related parameters. Only a few of the rank frequencies fall below the lower lines or are higher than the upper lines, and the shapes of the rank frequencies look random, suggesting no biases (see below for the types of biases simulation-based calibration can detect). The same holds true for all other parameters. These results are supported by the Pearson's chi-square statistic used to test for uniformity. About four percent of these statistics led to greater values than the critical value, which is expected with an α level of 0.05.

The results in Hartmann et al. (2020) suggest that the data-averaged posterior distributions equal the prior distributions. Therefore we do not have any evidence that the algorithm in Hartmann et al. (2020) has any problems. This indicates that the algorithm is indeed valid for the chosen models. Since we see no reason to expect other RT-MPT models would be much different we assume the algorithm to be valid for all RT-MPT models. Nevertheless, as Talts et al. (2018) argue, simulation-based calibration should be done for any model to be sure.

Simulation-based calibration is quite sensitive to the model. It can detect many types of biases. First, autocorrelation of the posterior samples. This means that there is a substantive correlation between two successive posterior samples. Second, over- and underdispersion. Overdispersion occurs when the posterior distribution is flatter

FIGURE 2.1

Histograms of the Rank Statistic for the Group-Level Process-Related Parameters



Note. This figure is copied from Hartmann et al. (2020). On the horizontal axes are the rank statistics and on the vertical axes the relative frequencies (frequencies divided by $N = 2000$). The term μ^{α} denotes the group-level parameters referring to the probabilities on the probit scale and μ^{β} the process rates. Lower and upper lines specify the critical values of the exact two-tailed binomial test with $\alpha = .05$ for deviations between the observed and the expected frequency (middle line) of each individual bin. D and g denote the processes and $-$ and $+$ the outcomes for the process rates.

than the prior distribution. Third, bias of the mean. This means that the mean of the posterior distribution is not at the same location as the mean of the prior distribibuiton.

For reasons of comparison, we fitted the same models to the data of Dube et al. (2012) and Arnold et al. (2014) as in Klauer and Kellen (2018) and compared the resulting medians and 95% highest density intervals. The medians were almost identical, but some of the highest density intervals were narrower in `rtmpt` compared to the results in Klauer and Kellen (2018). This might be due to smaller changes in the prior choices (for more details see Hartmann et al., 2020).

In addition to simulation-based calibration, we conducted smaller recovery studies. They provide also no evidence of the algorithm being wrong, i.e., the true values were recovered quite well in these studies.

Chapter 3

Extending RT-MPT

In this chapter I want to outline the issue with the first algorithm we used for RT-MPT modeling (Hartmann et al., 2020) and the way it was solved in Hartmann and Klauer (2020a)[1]. Since this required a modified algorithm we wanted to make sure the modified algorithm still works as well as the previous one.

3.1 Restriction on RT-MPTs

The algorithm in Hartmann et al. (2020) is based on a likelihood function that is not differentiable at any point – or rather the implementation of the likelihood function is not. It is constructed as the convolution of a hypo-exponential distribution and a truncated normal distribution. The hypo-exponential distribution, in turn, is a convolution of multiple exponential distributions with different rate parameters. Unfortunately, the density of this distribution includes at least one term where two rate parameters are subtracted in the denominator. If these two parameters coincide we divide by zero.

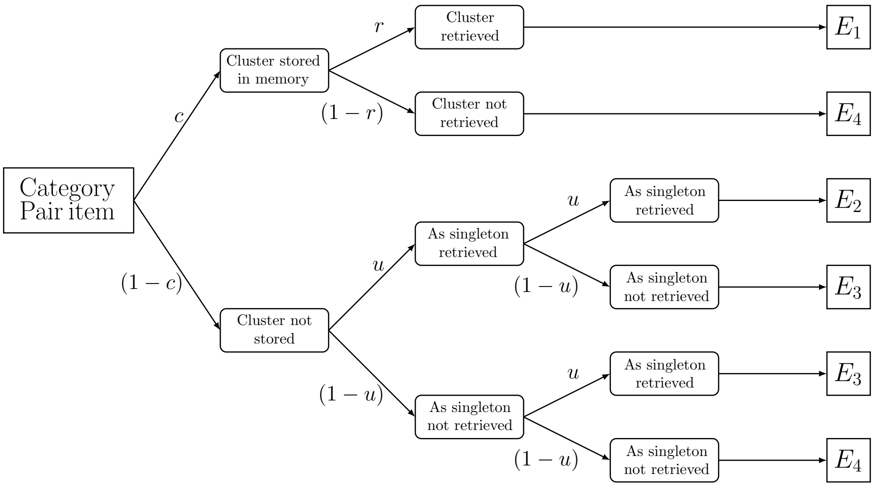
The consequence is that only RT-MPT models with distinct processes on each path of the (RT-)MPT tree work, i.e., models with repeated processes on one path cannot be used. One well known MPT model with repeated processes on a path is the *pair-clustering* model by Batchelder and Riefer (1980, 1986). According to Batchelder and Riefer (1980) this model can be used for *free-recall* tasks, where participants first study two types of words within a word list. One type of words is refered to as pairs (two words belonging to the same cluster) and one type of words is refered to as singletons (words belonging to no cluster). Figure 3.1 depicts the tree of this model where two clustered words need to be recalled. There are two paths in which a process is repeated, namely when the cluster is not stored; the process for a successful singleton-retrieval, u, and the process for a failed singleton-retrieval, $1-u$, are used twice.

Even though the pair-cluster model is a point in case, it is used in the context of free-recall tasks. This type of task does not allow for meaningful RT measures; it is not possible to tell when a participant is concerned with the task and which word he tries

[1] Raphael Hartmann is credited as the main author of this article. He helped integrate the new PDF of the RTs into the *C++* code, performed the simulations and analyses, derived an exemplary proof of the new formulae of the PDF, and wrote the first draft. Karl Christoph Klauer contributed by adjusting the *C++* code, provided suggestions for the simulations studies, advised the analyses, derived the new formulae of the PDF, and provided recommendations for structuring and improving the manuscript.

FIGURE 3.1

Paired-Words Tree for the Pair-Clustering MPT by Batchelder and Riefer (1980, 1986)



Note. "Category Pair item"-tree of Batchelder and Riefer's (1980, 1986) *pair-clustering* model. The model has two trees; One for clusterable paris ("Category Pair items"; two words that can be stored as a cluster) and one for singletons ("Singleton items"; words belonging to no cluster). The second one is not depicted here. Parameter c is the probability for storing a clusterble pair as cluster in memory, r the conditional probability that the stored cluster is retrieved from memory, and u the joint probability that a singleton is stored and retrieved. In the "Category Pair item"-tree u is the probability that a word from the clusterable pair is stored *as a singleton* and retrieved. E_1 is the response category for recalling both items as a pair, E_2 for recalling both items as singletons, E_3 for recalling only one of both items (as singleton), and E_4 for no recalling.

to remember. Therefore it is not possible to assign an RT to a specific word from the study phase.

Nevertheless, in general this type of model with repeating processes on one path typically arise whenever two or more items need to be evaluated at once. Hartmann and Klauer (2020a) uses variations of the 2HTM for *two-alternative forced-choice* tasks, where participants study a list of words in the study phase. In the test phase in each trial always two words are presented where one of which is a studied word and one is new. The task is to identify the studied word. Figure 3.2 depicts a possible MPT model for such a two-alternative forced-choice task.

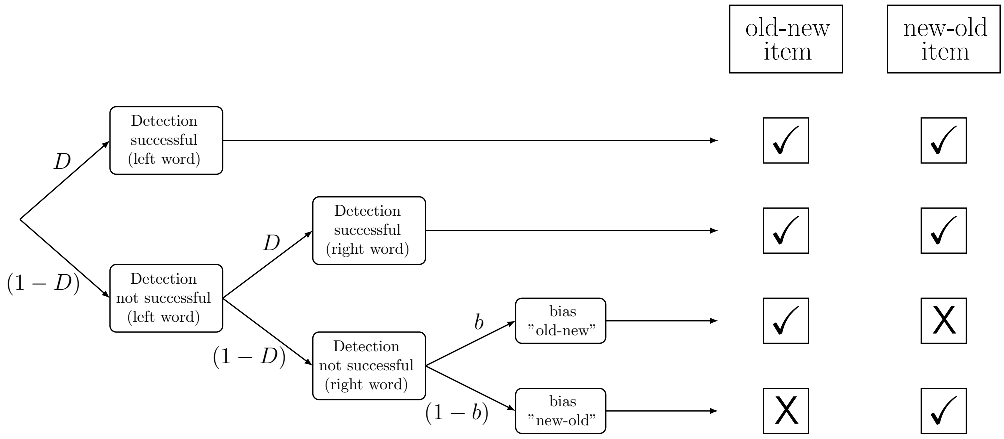
Since the algorithms by Klauer and Kellen (2018) and Hartmann et al. (2020) are restricted to models having no such repetitions of processes in their trees, it was desirable to develop a new algorithm with which RT-MPT models like the above mentioned can be fitted. This improves the applicability of the model class of RT-MPTs.

3.2 The Modified Algorithm

The main problem of developing an algorithm able to handle process repetitions within a tree of an RT-MPT model is the mix of processes with equal and unequal exponential rates. As already mentioned, if the exponential rates are all distinct the convolution is a

FIGURE 3.2

Tree Representation of an MPT Model for a 2AFC Task



Note. This figure is copied from Hartmann and Klauer (2020a). Parameter D refers to the detection of one word, and b to the bias towards the *"old-new"*-response ($1-b$ is the bias towards the *"new-old"*-response). The model assumes that as soon as one word is detected the corresponding response is given. Only if both words are not detected, will the bias determine the response. The columns on the right side indicate for each subtree/stimulus (*"old-new"* or *"new-old"* items) whether the branch leads to the correct answer or not.

hypo-exponential distribution with n different rates. The convolution with a (truncated) normal distribution has a closed form and was used in Klauer and Kellen (2018) and Hartmann et al. (2020). The convolution of n exponential distributions with equal rates is an gamma distribution with shape n (a.k.a. Erlang distribution with a shape parameter n). The convolution of a gamma distribution (with $n \in \mathbb{N}$) with a (truncated) normal distribution is not straight forward; there exists no closed-form expression for an arbitrary n. Nevertheless, using a specific n there exists a closed-form expression. In Hartmann and Klauer (2020a) such expressions are derived for $n \in \{2,3,4,5\}$.

In addition to that the convolution of exponential distributions with combinations of both cases (equal and unequal rates) was also needed. Scheuer (1988) as well as Amari and Misra (1997) provide a closed-form expression for the convolution of exponential distributions with a mix of equal and unequal rates. Jasiulewicz and Kordecki (2003) provide a proof of this formula.

Combining these two results Hartmann and Klauer (2020a) derived the convolution of exponential distributions (with equal and unequal rates) and a truncated normal distribution for cases when the number of equal rates equals $n \in \{1,2,3,4,5\}$. An RT-MPT model with more than five equal exponential rates is rather improbable, but could be implemented if needed.

3.3 Algorithm Validation

The newly developed algorithm in Hartmann and Klauer (2020a) was also validated using simulation-based calibration (Talts et al., 2018) like the algorithm in Hartmann et al. (2020). For more details about the core idea see Section 2.4 and for the exact procedure see Hartmann et al. (2020) as well as Hartmann and Klauer (2020a). The only difference to the algorithm validation in Hartmann et al. (2020) is the model. Hartmann and Klauer (2020a) used the 2HTM for two-alternative forced-choice tasks, where both presented words in a trial are tried to be detected (with probability D) and if both detection fail the studied word is guessed (with probability b). Figure 3.2 depicts the graphical representation of the model.

The results in Hartmann and Klauer (2020a) indicate that there was no problem with the modified algorithm. This suggests that the newly developed algorithm is valid, at least for the chosen altered 2HTM. We strongly believe that for other RT-MPT models the algorithm is valid as well, but according to Talts et al. (2018) this should be checked for each model separately.

Chapter 4

Basis for Modeling Processes With Wiener Diffusion Models

Even though the algorithm in Hartmann et al. (2020) seems to work in a satisfactory manner, it is still a restriction of the current RT-MPTs that the process-time and motor-time distributions are fixed to an exponential and a truncated normal distribution, respectively. In this chapter I want to discuss another possibility of modeling RTs. A basis for this is provided by Hartmann and Klauer (2020b)[1].

4.1 Process-Time Distribution

RT-MPTs (Hartmann et al., 2020; Hartmann & Klauer, 2020a; Klauer & Kellen, 2018) so far are restricted to the chosen process-time and motor-time distributions. As already mentioned this is due to the fact that convolutions of multiple distributions not always have a closed-form expression. Nevertheless, the exponential distribution might be not an optimal choice for a process time due to its properties. One such property is the *memorylessness*:

$$\mathbb{P}(T > s) = \mathbb{P}(T > s + t | T > t) = \frac{\mathbb{P}(T > s + t)}{\mathbb{P}(T > t)}, \tag{4.1}$$

where T is the process completion time, i.e., the probability for a process to complete after a certain time s is the same as the probability of this process to complete after time $s + t$, given it has not finished until time t. In other words it does not depend on whether the process just started or some time t has already passed without the process being finished, the probability that the processes completes after time s is always the same. This property might be unrealistic in most real processes. One would naturally assume this probability would change over time t.

[1] Raphael Hartmann is credited as the main author of this article. He helped deriving the partial derivatives, performed the simulations and analyses and wrote the first draft. Karl Christoph Klauer contributed by deriving the partial derivatives, provided suggestions for the simulations, and provided recommendations for structuring and improving the manuscript.

4.2 Alternative Process-Time Distributions

Klauer and Kellen (2018) suggested other process-time distributions like the Wald distribution, which is a first-passage time distribution of a Brownian motion with positive drift. One problem that might arise is the number of parameters used for each process. The exponential distribution has only one parameter but the Wald distribution has two.

Another option discussed by Klauer and Kellen (2018) is a diffusion model for modeling the process outcomes and process times together. For example, the Wiener diffusion model without non-decision time seems to be a good option. In RT-MPT by Klauer and Kellen (2018) for each process (outcome and times) three parameters are needed (two process rates and one process probability). When using the Wiener drift diffusion model without non-decision time to model the processes the number of required parameters woul be the same.

The Wiener diffusion model (e.g., Ratcliff, 1978; Wabersich & Vandekerckhove, 2014) is typically used in the context of decision tasks with two alternative choices. It provides an elegant way of modeling frequency and latency data jointly. On a conceptual level its core assumption is that the process of reaching a decision is an accumulation of evidence or information over time and stopping at one of two barriers where enough evidence for one or the other choice is gathered. The start z is somewhere between two barriers – a lower at 0 and an upper at a – and the tendency for the process to drift towards the upper barrier is denoted with v.

Figure 4.1 depicts such a model where the evidence accumulation leads to the upper barrier (black path), i.e., to the corresponding outcome of the process (e.g., a success in detecting a word). With this type of modeling processes it is only possible to use processes having exactly two outcomes. If a process has more the model must be transformed to fulfill this requirement, which is always possible.

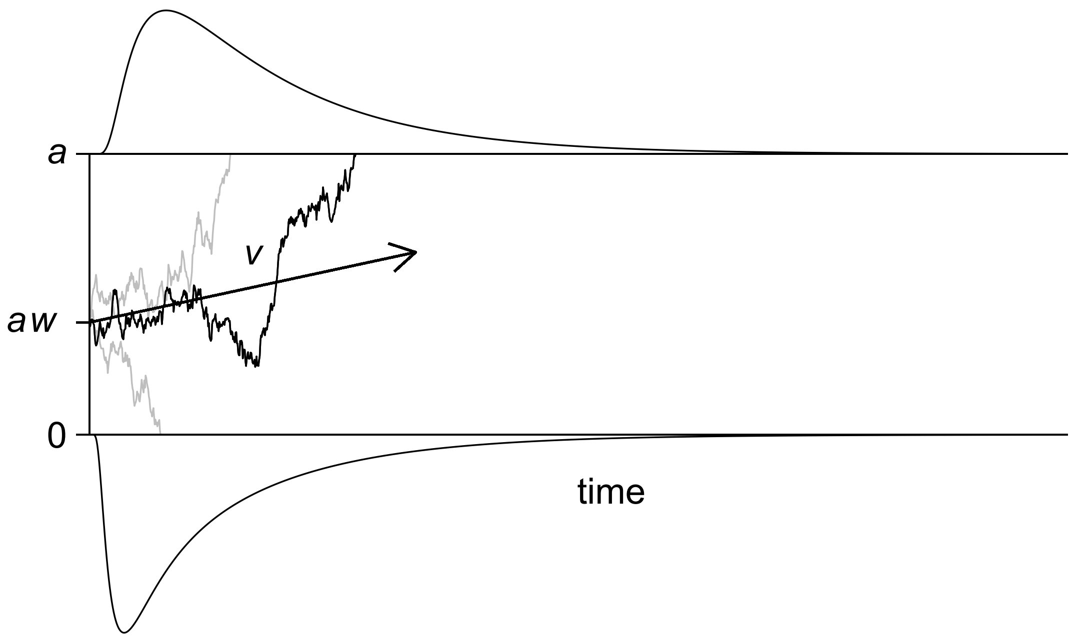
4.3 Use of Density/Likelihood Functions

One problem already mentioned is the lack of a closed-form expression for the density of a convolution in many cases. The density function of the RTs is needed in the MCMC algorithm in order to use data augmentation for a more efficient Gibbs sampler (Albert & Chib, 1993). In this data augmentation step the density for each possible path leading to the same response category (e.g., responding “old” to an old word by detection or by guessing) is compared by using ratios (for more details see Klauer, 2009; Klauer & Kellen, 2018). Therefore, one can assign a probability to every path and sample one of them in each MCMC iteration step. The likelihood functions are also needed for the calculation of information criteria, like the deviance information criterion (Spiegelhalter et al., 2002).

Since it is not possible to find a closed-form expression for a convolution of multiple first-passage time distributions (and a motor-time distribution), another approach is

FIGURE 4.1

Graphic Representation of a Wiener Diffusion Model for a Potential Process



Note. The upper barrier is denoted with a, the relative starting point with w, and the drift rate with v (depicted as an arrow). The lower barrier is denoted with 0. Above the upper barrier and below the lower barrier the density for the corresponding responses – or rather outcomes in the context of processes – is sketched. The black jittery line starting at aw and ending at the upper barrier represents an exemplary evidence accumulation (random walk). The grey lines represent alternative exemplary evidence accumulations, one for the same outcome and one for the other.

needed. Carlin and Chib (1995) provide a promising approach to solve the problem for data augmentation. The idea is to treat each path as a separate model and calculate Bayes factors for these models, which are nothing more than ratios. Therefore, one can assign probabilities to the different paths leading to the same response category and sample one of these paths in each MCMC iteration step.

This promising approach by Carlin and Chib (1995) cannot solve the problem of calculating the information criteria. Therefore, model selection might not be possible or would require a numerical approach.

4.4 Basis for Modeling Processes With a Wiener Diffusion Model

The first-passage time PDF and CDF of the Wiener diffusion model can only be expressed as infinite series (Blurton et al., 2012; Navarro & Fuss, 2009). For practical reasons of computation these series need to be truncated at some point, i.e., only a specific number of components is calculated. Blurton et al. (2012), Gondan et al. (2014), and Navarro and Fuss (2009) derived upper bounds for the absolute approximation error, incurred by truncation, and for the number of components that is required to guarantee a predetermined precision.

Unfortunately, calculating multiple components of a series can slow down computation. Therefore, it is even more important to have efficient optimization routines. Feeding these routines with gradient information increases their efficiency.

There exist two different infinite-series expressions for the PDFs and CDFs – one of which converges faster for small first-passage times and one converges faster for large first-passage times. In Hartmann and Klauer (2020b) we derived the partial derivatives of the first-passage time PDF and CDF for both series expressions. Since already the PDF and CDF can only be represented as infinite series, so can the partial derivatives thereof. Therefore we derived upper bounds for the absolute approximation error and for the number of components that is required to guarantee a predetermined precision. In addition Hartmann and Klauer (2020b) developed an *R* package for the calculation of the partial derivatives

Chapter 5

Discussion

We successfully accomplished the three main goals. First, we implemented the RT-MPT model class into *R* by developing an *R* package, called `rtmpt`. In addition, we demonstrated that the algorithm underlying `rtmpt` is suitable for an RT-MPT model and that the *R* package provides similar parameter estimation compared to the ones in Klauer and Kellen (2018). Second, we extended the model class to enable the fit of RT-MPT models with up to five identical processes on a path – or rather up to five equal process times on a path. In addition, we demonstrated that the newly developed algorithm for `rtmpt` that enables this is valid for an RT-MPT model. Third, we derived the partial derivatives of the first-passage time PDF and CDF of the Wiener diffusion model and developed an *R* package to use these derivatives. These derivatives can be seen as a basis for efficiently modeling the processes within RT-MPT with Wiener diffusion models. There are other related projects that we will work on or might be interesting to do in the future.

One of the projects that we are working on concerns the robustness of the RT-MPT estimation against the violation of the distributional assumptions underlying the process times and motor times. We try to investigate how robust the estimates are when using different process-time and motor-time distributions. As process-time distributions we use the gamma distribution with a shape parameter of two (a.k.a. Erlang distribution with shape of two), the Wald distribution (a.k.a inverse Gauss distribution), and the first-passage time distribution of the Wiener diffusion model without non-decision time. As a control we use the exponential distribution, which is the assumed process-time distribution in RT-MPT (Hartmann et al., 2020; Hartmann & Klauer, 2020a; Klauer & Kellen, 2018). As motor-time distributions we use Student's t distributions truncated from below at zero and with different degrees of freedom (5 and 15). As a control we use the normal distribution truncated from below at zero. So far the results are quite mixed for the non-control conditions. Nevertheless, the results suggest that using different process-time distributions in RT-MPT might be desirable.

Another project we have in mind is a sampling procedure to efficiently sample from the first-passage time distribution of the Wiener diffusion model. In this project we will use an adaptive sampling method based on Gilks and Wild (1992). For that the partial derivative of the first-passage time PDF of the Wiener diffusion model with respect to t is required.

A more applied project might be a follow up on our paper Hartmann and Klauer (2020a). In this paper we use the data from Province and Rouder (2012) with an RT-MPT model based on an altered 2HTM to fit data from a two-alternative forced-choice task. The task included a word-strength manipulation leading to three different conditions: A target word was either seen once in the study phase, twice, or four times. We used this data only to demonstrate the usefulness of our newly developed algorithm. Nevertheless, Province and Rouder (2012) claimed that the mean RTs are conditionally independent, i.e., conditioned on the mental state the mean RTs do not depend on the stimulus condition. A mental state in this case might be the detection of the target word. The stimulus condition in this case refers to the word-strength manipulation. According to our findings in Hartmann and Klauer (2020a) we could not support Province and Rouder's (2012) conclusion. It would be interesting to further investigate conditional independence with different RT-MPT models and/or different manipulations (e.g., base-rate manipulation).

Mostly, the RT-MPT model class is applicable whenever the MPT model is applicable. Erdfelder et al. (2009), for example, gives a broad overview of the psychological fields in which MPT models are used. They name the MPT models typically used for each field. Nevertheless, there are some restrictions to RT-MPT models. They can only be applied whenever RTs are available. For example, when using a free-recall task we usually do not have meaningful RTs. Therefore, extending the pair-clustering model (Batchelder & Riefer, 1980, 1986) to an RT-MPT and applying it to a free-recall task would not be feasible. The advantage of RT-MPT, thought, is that many unidentified MPTs become identified once the models are extended to RT-MPTs (Klauer & Kellen, 2018).

With its possibility to estimate process times the RT-MPT model class has the potential to provide more insights into the cognitive mechanisms underlying the completion of decision tasks. It allows for testing the order in which cognitive processes take place as well as whether the completion time of two process outcomes is the same. Next to questions from cognitive research RT-MPT models might also be of interest for research about mental disorders. It could, for example, give an answer to whether there are differences in the process times between a group of patients with a specific mental disorder and a control group. Nevertheless, RT-MPT as it is implemented now cannot be used as a diagnostic tool because of its hierarchical structure; The algorithm expects more than one subject.

Next to different process-time and motor-time distributions, as discussed in Chapter 4, there are a few extensions one might want to introduce to RT-MPT. One of these extensions could be the inclusion of a Bayes factor – or rather the calculation of the marginal likelihood (the probability of the data given the model) – in order to have an alternative method for model selection (e.g., Gelman et al., 2013) to the information criteria already provided in Hartmann and Klauer (2020a). Bayes factors are analogous to likelihood ratio tests in classical statistics. The Bayes factor is calculated by taking the ratio of the marginal likelihoods of two models, $p(y|M_2)/p(y|M_1)$, and if both models have a priori the same probability, then the Bayes factor coincides with the probability of the second model, M_2, given the data divided by the probability of the first model, M_1, given the data $p(M_2|y)/p(M_1|y)$ (Gelman et al., 2013).

Another interesting extension might be the possibility to set process times equal – or rather their distribution. So far, process times are only set equal if the underlying processes are the same. For two different processes the process times differ, in general. It might be of interest to set two process times equal, for example, when using a model with different detection processes that are assumed to take the same time but have different probabilities for successfully finishing. Suppose we have words from different sources and these words need to be recalled. It would be a plausible assumption that the probability for recalling words from different sources can differ but the time with which they are recalled is the same. Or at least it might be interesting to test this assumption. Another example would be when participants are required to give judgements about the certainty with which they remember a word. Even though the probabilities for the different judgements might differ, it is plausible to assume that the time to decide is the same for all judgements.

References

Albert, J. H., & Chib, S. (1993). Bayesian analysis of binary and polychotomous response data. *Journal of the American Statistical Association*, *88*, 669–679. https://doi.org/10.1080/01621459.1993.10476321

Amari, S., & Misra, R. (1997). Closed-form expressions for distribution of sum of exponential random variables. *IEEE Transactions on Reliability*, *46*, 519–522. https://doi.org/10.1109/24.693785

Arnold, N. R., Bröder, A., & Bayen, U. J. (2014). Empirical validation of the diffusion model for recognition memory and a comparison of parameter-estimation methods. *Psychological Research*, *79*, 882–898. https://doi.org/10.1007/s00426-014-0608-y

Batchelder, W. H., & Riefer, D. M. (1980). Separation of storage and retrieval factors in free recall of clusterable pairs. *Psychological Review*, *87*, 375–397. https://doi.org/10.1037/0033-295x.87.4.375

Batchelder, W. H., & Riefer, D. M. (1986). The statistical analysis of a model for storage and retrieval processes in human memory. *British Journal of Mathematical and Statistical Psychology*, *39*, 129–149. https://doi.org/10.1111/j.2044-8317.1986.tb00852.x

Batchelder, W. H., & Riefer, D. M. (1999). Theoretical and empirical review of multinomial process tree modeling. *Psychonomic Bulletin & Review*, *6*, 57–86. https://doi.org/10.3758/bf03210812

Blurton, S. P., Kesselmeier, M., & Gondan, M. (2012). Fast and accurate calculations for cumulative first-passage time distributions in Wiener diffusion models. *Journal of Mathematical Psychology*, *56*, 470–475. https://doi.org/10.1016/j.jmp.2012.09.002

Carlin, B. P., & Chib, S. (1995). Bayesian model choice via markov chain monte carlo methods. *Journal of the Royal Statistical Society: Series B (Methodological)*, *57*, 473–484. https://doi.org/10.1111/j.2517-6161.1995.tb02042.x

Donders, F. C. (1868). Over de snelheid van psychische processen. *Onderzoekingen gedaan in het Physiologisch Laboratorium der Utrechtsche Hoogeschool (1968–1869)*, *2*, 92–120.

Donders, F. C. (1969). On the speed of mental processes (W. G. Koster, Trans.) *Acta Psychologica*, *30*, 412–431. https://doi.org/10.1016/0001-6918(69)90065-1 (Original work published 1868)

Dube, C., Starns, J. J., Rotello, C. M., & Ratcliff, R. (2012). Beyond ROC curvature: Strength effects and response time data support continuous-evidence models of recognition memory. *Journal of Memory and Language*, *67*, 389–406. https://doi.org/10.1016/j.jml.2012.06.002

Erdfelder, E., Auer, T.-S., Hilbig, B. E., Aßfalg, A., Moshagen, M., & Nadarevic, L. (2009). Multinomial processing tree models. *Zeitschrift für Psychologie / Journal of Psychology*, *217*, 108–124. https://doi.org/10.1027/0044-3409.217.3.108

Galassi, M., Davies, J., Theiler, J., Gough, B., Jungman, G., PatricAlken, Booth, M., & Rossi, F. (Eds.). (2009). *GNU scientific library reference manual* (3rd ed.). Network Theory Ltd.

Gelfand, A. E., & Smith, A. F. M. (1990). Sampling-based approaches to calculating marginal densities. *Journal of the American Statistical Association*, *85*, 398–409. https://doi.org/10.1080/01621459.1990.10476213

Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., & Rubin, D. B. (2013). *Bayesian data analysis* (3rd ed.). Chapman; Hall/CRC. https://doi.org/10.1201/b16018

Gilks, W. R., & Wild, P. (1992). Adaptive rejection sampling for gibbs sampling. *Applied Statistics*, *41*, 337. https://doi.org/10.2307/2347565

Gondan, M., Blurton, S. P., & Kesselmeier, M. (2014). Even faster and even more accurate first-passage time densities and distributions for the Wiener diffusion model. *Journal of Mathematical Psychology*, *60*, 20–22. https://doi.org/10.1016/j.jmp.2014.05.002

Hartmann, R., Johannsen, L., & Klauer, K. C. (2020). Rtmpt: An r package for fitting response-time extended multinomial processing tree models. *Behavior Research Methods*, *52*, 1313–1338. https://doi.org/10.3758/s13428-019-01318-x

Hartmann, R., & Klauer, K. C. (2020a). Extending RT-MPTs to enable equal process times. *Journal of Mathematical Psychology*, *96*, 102340. https://doi.org/10.1016/j.jmp.2020.102340

Hartmann, R., & Klauer, K. C. (2020b). *Partial derivatives for the first-passage time distribution in Wiener diffusion models*. Manuscript submitted for publication.

Heck, D. W., & Erdfelder, E. (2016). Extending multinomial processing tree models to measure the relative speed of cognitive processes. *Psychonomic Bulletin & Review*, *23*, 1440–1465. https://doi.org/10.3758/s13423-016-1025-6

Heck, D. W., Erdfelder, E., & Kieslich, P. J. (2018). Generalized processing tree models: Jointly modeling discrete and continuous variables. *Psychometrika*, *83*, 893–918. https://doi.org/10.1007/s11336-018-9622-0

Hu, X. (2001). Extending general processing tree models to analyze reaction time experiments. *Journal of Mathematical Psychology*, *45*, 603–634. https://doi.org/10.1006/jmps.2000.1340

Hütter, M., & Klauer, K. C. (2016). Applying processing trees in social psychology. *European Review of Social Psychology*, *27*, 116–159. https://doi.org/10.1080/10463283.2016.1212966

Jasiulewicz, H., & Kordecki, W. (2003). Convolutions of erlang and of pascal distributions with applications to reliability. *Demonstratio Mathematica*, *36*(1), 231–238. https://doi.org/10.1515/dema-2003-0125

Klauer, K. C. (2009). Hierarchical multinomial processing tree models: A latent-trait approach. *Psychometrika*, *75*, 70–98. https://doi.org/10.1007/s11336-009-9141-0

Klauer, K. C., & Kellen, D. (2018). RT-MPTs: Process models for response-time distributions based on multinomial processing trees with applications to recognition memory. *Journal of Mathematical Psychology*, *82*, 111–130. https://doi.org/10.1016/j.jmp.2017.12.003

Krantz, D. H. (1969). Threshold theories of signal detection. *Psychological Review*, *76*, 308–324. https://doi.org/10.1037/h0027238

Link, S. W. (1982). Correcting response measures for guessing and partial information. *Psychological Bulletin*, *92*, 469–486. https://doi.org/10.1037/0033-2909.92.2.469

Luce, R. D. (1963). A threshold theory for simple detection experiments. *Psychological Review*, *70*, 61–79. https://doi.org/10.1037/h0039723

Matzke, D., & Wagenmakers, E.-J. (2009). Psychological interpretation of the ex-gaussian and shifted wald parameters: A diffusion model analysis. *Psychonomic Bulletin & Review*, *16*, 798–817. https://doi.org/10.3758/pbr.16.5.798

Navarro, D. J., & Fuss, I. G. (2009). Fast and accurate calculations for first-passage times in Wiener diffusion models. *Journal of Mathematical Psychology*, *53*, 222–230. https://doi.org/10.1016/j.jmp.2009.02.003

Plummer, M. (2018). Rjags: Bayesian graphical models using MCMC. http://mcmc-jags.sourceforge.net

Province, J. M., & Rouder, J. N. (2012). Evidence for discrete-state processing in recognition memory. *Proceedings of the National Academy of Sciences*, *109*, 14357–14362. https://doi.org/10.1073/pnas.1103880109

R Core Team. (2020). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. Vienna, Austria. https://cran.r-project.org/doc/manuals/fullrefman.pdf

Ratcliff, R. (1978). A theory of memory retrieval. *Psychological Review*, *85*, 59–108. https://doi.org/10.1037/0033-295x.85.2.59

Riefer, D. M., & Batchelder, W. H. (1988). Multinomial modeling and the measurement of cognitive processes. *Psychological Review*, *95*, 318–339. https://doi.org/10.1037/0033-295x.95.3.318

Scheuer, E. (1988). Reliability of an m-out of-n system when component failure induces higher failure rates in survivors. *IEEE Transactions on Reliability*, *37*, 73–74. https://doi.org/10.1109/24.3717

Schmittmann, V. D., Dolan, C. V., Raijmakers, M. E. J., & Batchelder, W. H. (2010). Parameter identification in multinomial processing tree models. *Behavior Research Methods*, *42*, 836–846. https://doi.org/10.3758/brm.42.3.836

Snodgrass, J. G., & Corwin, J. (1988). Pragmatics of measuring recognition memory: Applications to dementia and amnesia. *Journal of Experimental Psychology: General*, *117*, 34–50. https://doi.org/10.1037/0096-3445.117.1.34

Spiegelhalter, D. J., Best, N. G., Carlin, B. P., & van der Linde, A. (2002). Bayesian measures of model complexity and fit. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *64*, 583–639. https://doi.org/10.1111/1467-9868.00353

Stan Development Team. (2018). RStan: The R interface to Stan. http://mc-stan.org

Swets, J. A. (1961). Is there a sensory threshold?: When the effects of the observer's response criterion are isolated, a sensory limitation is not evident. *Science*, *134*, 168–177. https://doi.org/10.1126/science.134.3473.168

Talts, S., Betancourt, M., Simpson, D., Vehtari, A., & Gelman, A. (2018). Validating bayesian inference algorithms with simulation-based calibration. *arXiv preprint arXiv:1804.06788*.

Tanner, M. A., & Wong, W. H. (1987). The calculation of posterior distributions by data augmentation. *Journal of the American Statistical Association*, *82*, 528–540. https://doi.org/10.1080/01621459.1987.10478458

Wabersich, D., & Vandekerckhove, J. (2013). Extending JAGS: A tutorial on adding custom distributions to JAGS (with a diffusion model example). *Behavior Research Methods*, *46*, 15–28. https://doi.org/10.3758/s13428-013-0369-3

Wabersich, D., & Vandekerckhove, J. (2014). The RWiener package: An R package providing distribution functions for the Wiener diffusion model. *The R Journal*, *6*, 49–56. https://doi.org/10.32614/rj-2014-005

Appendix A

German Summary (Deutsche Zusammenfassung)

Die Modellklasse der multinomialen Verarbeitungsbäume (MPT) ist sehr bekannt in der Kognitionspsychologie. Mit Modellen dieser Klasse versucht man, kognitive Mechanismen, die Entscheidungsaufgaben zugrundeliegen, zu erklären. Solche Mechanismen beinhalten kognitive Prozesse und beschreiben, wie diese miteinander interagieren. Mithand der Antworthäufigkeiten ist es möglich, die Wahrscheinlichkeiten für Prozessausgänge zu schätzen.

Die Modellklasse der reaktionszeiterweiterten MPTs (RT-MPT; Klauer & Kellen, 2018) erweitert die MPT-Modellklasse, indem Reaktionszeiten zusätzlich zu den Antworthäufigkeiten integriert werden. Diese erlaubt es, Prozess-Beendigungszeiten – die Zeit bis ein Prozess in einem bestimmten Ausgang endet – und die Enkodierungs- und Motorausführungszeit zu schätzen. Jeder Prozess hat zwei mögliche Ausgänge und jeder mögliche Prozessausgang wiederum hat eine zugeordnete Wahrscheinlichkeit und Prozess-Beendigungszeit. Die erste Implementierung von RT-MPT war in *C++*, was es eher unzugänglich machte für potenzielle Nutzer, mal abgesehen von den Entwicklern, und beinhaltete eine proprietäre Programmbibliothek. Deshalb war die Implementierung weder einfach zu nutzen, noch kostenlos zu erhalten.

In dieser Dissertation erleichtere ich die Nutzung der RT-MPT-Modellklasse. In drei wissenschaftlichen Artikeln entwickeln ich und meine Mitautoren und -autorinnen ein Softwarepaket, das einfach zu nutzen, kostenlos und quelloffen ist, wir erweitern das Paket, um noch mehr RT-MPT-Modelle fitten zu können, wir validieren den Algorithmus des Softwarepakets, und wir legen eine Basis, um RT-MPT-Modelle, die einen Wiener Diffusionsprozess annehmen, effizient modelieren zu können. In unserem ersten Projekt integrieren wir die RT-MPT-Modellklasse in die Programmiersprache *R*, indem wir ein sogenanntes *R*-Paket entwickeln. Ebenso validieren wir das resultierende Softwarepaket und testen, ob es die gleichen Resultate liefert wie im originalen Artikel von Klauer and Kellen (2018). Im zweiten Projekt erweitern wir die Modellklasse – bzw. ihre Implementierung – so, dass RT-MPT-Modelle mit gleichen Prozessen auf einem Pfad gefittet werden können. Aufgrund der Eigenschaften der vorherigen Likelihood-Funktion war das bisher nicht möglich. Da sich der Algorithmus nun verändert hat, validieren wir auch die neue Implementierung, um zu schauen, ob der neue Algorithmus richtig funktioniert. Im letzten Projekt leiten wir die partiellen Ableitungen der Dichte- und Verteilungsfunktion der Absorptionszeit im Wiener Diffusionsmodell her. Wie Klauer and Kellen (2018) andeuten, wäre das Wiener Diffusionsmodell eine geeignete Option für die Modellierung von Prozessen. Es wäre eine konkurierende Alternative zur momentanen Implementierung, in der die Prozess-Beendigungszeiten exponentialverteilt sind (Klauer & Kellen, 2018). Partielle Ableitungen liefern Informationen, die für effiziente Methoden der Parameterschätzung genutzt werden könnnen.