A Fast Numerical Algorithm for the Estimation of Diffusion Model Parameters

Andreas Voss[*]
Albert-Ludwigs-Universität Freiburg
Department of Psychology
79085 Freiburg
Germany

Jochen Voss[†]
University of Warwick
Mathematics Institute
Coventry, CV4 7AL
UK

October 25, 2007

Abstract

In this paper, we describe a new algorithmic approach for parameter estimation in Ratcliff's (1978) diffusion model. This problem, especially if inter-trial variabilities of parameters are included in the model, is computationally very expensive; the parameter estimation procedure often takes a long time even with today's high-speed computers. The algorithm described here makes the calculation of the cumulative distribution functions for predicted process durations computationally much less expensive. This improvement is achieved by solving the Kolmogorov backward equation numerically instead of employing the previously used closed form solution. Additionally, the algorithm can determine the optimum fit for one of the model parameters (the starting point z) directly, thereby reducing the dimension of the parameter search space by one. The resulting method is shown to be notably faster than the standard (closed-form solution) method for parameter estimation.

1 Introduction

The diffusion model was proposed for the analysis of fast binary decisions by Roger Ratcliff (e.g., Ratcliff, 1978; Ratcliff, Van Zandt, and McKoon, 1999) nearly three decades ago. Since then it has been applied in different domains of cognitive psychology like cognitive aging (e.g., Ratcliff, Spieler, and McKoon, 2000; Ratcliff, Thapar, Gomez, and McKoon, 2004; Ratcliff, Thapar, and McKoon, 2001, 2003), memory retrieval (e.g., Ratcliff, 1978; Spaniol, Madden, and Voss, 2006), and perceptional processes (e.g. Ratcliff, 2002; Ratcliff, Thapar, and McKoon, 2001; Voss, Rothermund, and Brandtstädter, in press; Voss, Rothermund, and Voss, 2004). The great benefit of this kind of modelling lies in the diffusion model's capacity to extract maximal information from a given set of data. Unlike traditional methods in experimental psychology which normally only use either mean response times or accuracy data, a diffusion model analysis is based on the full shape of the response time distribution for correct responses and errors and—simultaneously—on the proportion of error responses. This exhaustive use of information allows detailed conclusions about the cognitive processes that cause the empirical response time distribution (Voss et al., 2004).

However, this benefit of the diffusion model comes at a high cost: a lot of computational power is needed to estimate the model parameters. Calculation is especially expensive when the complete model as proposed by Ratcliff (e.g., Ratcliff and Rouder, 1998; Ratcliff and Tuerlinckx, 2002) is implemented; in this model, parameter variability is allowed across trials of an experiment. This makes computation so expensive that—even with a fast computer—estimation of the model parameters might take several hours or even days. The problem of long computation times is especially pointed when parameters are calculated separately for each participant and

[*] voss@psychologie.uni-freiburg.de

[†] Jochen.Voss@warwick.ac.uk

when the Kolmogorov-Smirnov method (Voss et al., 2004) or the maximum likelihood estimator are used rather than χ^2-based algorithms (Ratcliff and Tuerlinckx, 2002).

The dominant contribution to the computation time comes from the computation of the predicted RT-distribution which forms the basis of the parameter estimation procedure (see eq. (2) in section 4, *The PDE Method*; cf. also: Ratcliff, 1978; Voss et al., 2004): In the process a sum with a—theoretically—infinite number of terms has to be evaluated frequently. Voss et al. (2004, Appendix B) show how the sum can be truncated to reach a given accuracy. However, the problem of huge computation times is still present.

We propose an alternative approach to solve this problem. While so far the diffusion model has always been calculated by eq. (2), there is a less expensive method: the predicted RT-distribution can be regarded as the solution of a partial differential equation (PDE). Instead of solving this equation analytically and reaching eq. (2) with its infinite sum, it is also possible to solve the equation numerically. While yielding the same accuracy, the PDE method is substantially faster than the better known 'closed form' solution.

We start the presentation, in Section 2, by giving a short overview over the diffusion model. Section 3 discusses the required parameter estimation procedure. Section 4 contains the central result of this article: it explains how the predicted RT-distribution can be computed as part of the algorithm by numerically solving a PDE. Section 5 compares the resulting algorithm with previously suggested methods and Section 6 contains some concluding remarks.

Finally, in two appendices, we present some mathematical background material for the proposed method: Appendix A contains a complete proof that the predicted RT-distribution can indeed be found as the solution to the given PDE. And Appendix B contains a short primer about numerical solution of PDEs, including pointers to the relevant literature.

Our own parameter-estimation software (*fast-dm*), implementing the method introduced in this paper, is described in detail in Voss and Voss (in press) and is available for download from the authors' homepages[1].

2 Description of the Diffusion Model

A diffusion model analysis is adequate for nearly any data from speeded binary decisions, where "speeded" means mean response times up to about one or two seconds. The diffusion model is based on a Brownian Motion with constant drift (see Figure 1): this diffusion process represents an internal counter on which information in a trial of a task is accumulated. The process runs between two thresholds and is terminated as soon as one of the thresholds is hit. The process duration represents the duration of the decisional process and position of the process at the end (i.e., the threshold that has been hit) represents the outcome, that is, the decision that has been reached.

The diffusion model is described by a number of parameters (e.g., Voss et al., 2004). Firstly, the process is driven by a systematic drift (denoted by v) and random fluctuations. In psychological terms the drift, which is assumed to be constant over time, is a measure of the participant's performance in the given task. For example, in a perceptual situation it represents the ability to distinguish between two classes of stimuli. The random fluctuations add noise to the process and lead to different process paths in different trials of an experiment. Therefore, processes with the same drift rate vary in duration, and processes may even end at the threshold opposite to the drift direction. The amount of noise s can be regarded as a model parameter called the diffusion constant (this is sometimes referred to as the "intra-trial variability of the drift"). However, s is only a scaling parameter, that is, changes in s just re-scale all other parameters linearly. For the purpose of estimating the model parameters we fix the diffusion constant to the value 1.

The second model parameter is the threshold separation a. The larger this value, the longer the process runs on average. At the same time accuracy of the decision process is increased by large values of a, that is, the process stops more often at the threshold corresponding to the sign of the drift (i.e., the upper threshold, if the drift is positive, and vice versa). Therefore, the threshold separation can be regarded as a measure of conservatism.

The third parameter of the model is the starting point z. When the same amount of information is needed before reaching the alternative decisions, z will equal $a/2$. In a biased situation,

[1] `http://www.psychologie.uni-freiburg.de/Members/voss/fast-dm`

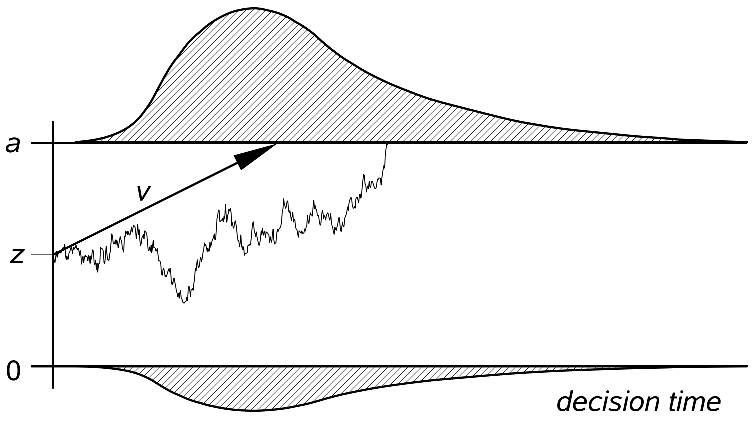

Figure 1: A schematic sketch of the Diffusion Model. A diffusion process with constant drift v starts at z and is terminated as soon as it reaches one of the thresholds at a or 0, respectively.

for example when pay-off matrices make one response more attractive, the starting point may be shifted towards one of the thresholds (Voss et al., 2004, in press).

The diffusion model describes just the decision process, which accounts only for a portion of the total response time. The remaining part of the response time comprises encoding as well as response processes. In the model all non-decisional processes are combined into the so-called RT constant (denoted by t_0 or sometimes t_{er}, respectively).

In typical psychological research situations, there is often substantial variation of parameter values between different trials of an experiment. Ratcliff (Ratcliff and Rouder, 1998; Ratcliff and Tuerlinckx, 2002) showed that model fit can be substantially improved, when this inter-trial variability is explicitly modelled. More specific, the complete diffusion model allows for inter-trial variability of drift rate (s_v or sometimes η), starting point (s_z), and of the non-decisional component (s_{t_0}). The actual drift rate in one trial of a task is then assumed to be normally distributed with mean v and standard deviation s_v. For the sake of simplicity, the actual starting point is assumed to have a uniform distribution from $z - 0.5\,s_z$ to $z + 0.5\,s_z$. Likewise, a uniform distribution is assumed for the actual RT-constant, ranging from $t_0 - 0.5\,s_{t_0}$ to $t_0 + 0.5\,s_{t_0}$.

3 Parameter Estimation

The diffusion model data analysis basically can be described as a parameter-search problem: in order to apply the diffusion model from section 2 in a psychological context a set of parameters has to be found so that the predicted response time distributions of the model optimally fit the empirical response times. To solve this problem a measure of the fit between the two distributions has to be chosen. Commonly used methods minimise the Maximum-Likelihood (ML) or the χ^2 statistics (Ratcliff and Tuerlinckx, 2002). More recently the use of the Kolmogorov-Smirnov (KS) statistic was suggested (Voss et al., 2004). In our view using the KS statistic is especially promising because it does not lose information by aggregating data (as the χ^2 statistic does) and it is not as strongly affected by outliers in the RT distribution as the ML statistic. Thus, in the remainder of this paper, we will mainly address the KS method. However, the numerical approach presented below can also be used for χ^2 methods.

The KS statistic is the maximum vertical distance between the predicted and the empirical cumulative response time distributions. If the KS-statistic is used for the diffusion model analysis, two cumulative distribution functions (CDFs)—for the two alternative responses—have to be considered simultaneously. This problem is solved by merging both CDFs together, the "error distribution" is formally mapped to the negative axis. This procedure is described in more detail in Voss et al. (2004). A second—more practical—problem arises from the fact that the predicted CDF has to be calculated very often, namely for all measured response times. In comparison, the χ^2 method requires values of the CDF only for the boundaries of the chosen RT bins. This makes use of the KS-statistic more costly compared to use of the χ^2 statistic.

Once an optimisation criterion is defined, a multidimensional search procedure can be employed to find the optimal set of parameters. For this purpose we use the downhill simplex method of Nelder and Mead (1965). Since the method works best when an approximation to the optimum is already known, the initial simplex is constructed from the result of the EZ-diffusion procedure (Wagenmakers, van der Maas, and Grasman, 2007). More details about parameter estimation procedures may be found elsewhere (e.g., Ratcliff and Tuerlinckx, 2002; Ratcliff et al., 1999; Voss et al., 2004).

4 The PDE Method

In order to implement the algorithm described in the previous section we need to compute the cumulative distribution function of the response-time distribution as given by the diffusion model from section 2. In this section we derive a partial differential equation whose solution is the CDF in question. Using standard methods to numerically solve this PDE leads then to efficient algorithms for computing the CDF and thus for solving the parameter estimation problem described in section 3.

Let F_+ be the probability that a Brownian Motion with constant drift v and starting at z hits $a > z$ before time t and before the first visit at 0. Let F_- be the corresponding probability for exit through 0 before hitting a. The (defective) CDFs F_+ and F_- can be found with the help of the Kolmogorov backward equation, described in Feller (1971) and, more explicitly, in equations (2.7) and (2.8) of Grasman and van Herwaarden (1999): F_+ is the solution of the PDE

$$\frac{\partial}{\partial t}F_+(t,z) = \frac{1}{2}\frac{\partial^2}{\partial z^2}F_+(t,z) + v\frac{\partial}{\partial z}F_+(t,z) \quad \text{for all } t > 0,\ 0 < z < a \tag{1a}$$

with boundary conditions

$$F_+(t,0) = 0, \quad F_+(t,a) = 1 \qquad \text{for all } t > 0 \tag{1b}$$

and initial condition

$$F_+(0,z) = \begin{cases} 0 & \text{if } 0 \leq z < a \text{ and} \\ 1 & \text{if } z = a. \end{cases} \tag{1c}$$

A complete proof of this fact can be found in Appendix A.

Previous algorithms to estimate parameters of the diffusion model were based on the observation that the solution F_+ of this PDE can be found explicitly: it is given by

$$F_+(t,z) = 2\pi \mathrm{e}^{(a-z)v} \sum_{k=1}^{\infty} k \sin\Bigl(\frac{\pi(a-z)k}{a}\Bigr) \frac{1 - \mathrm{e}^{-\frac{a^2v^2+\pi^2k^2}{2a^2}t}}{a^2v^2 + \pi^2k^2} \tag{2}$$

(Ratcliff (1978); Voss et al. (2004), cf. also Section XIV.5 of Feller (1971) for a description of how (2) can be derived from (1)).

The CDF F_- for the lower boundary can be obtained by using drift $-v$ (instead of v) and starting point $a - z$ (instead of z) in either Formula (1) or (2).

While the closed solution (2) is satisfying from a mathematical point of view, evaluating it numerically is expensive. And to include variability in the parameters it is necessary to integrate F_+ and F_- over z, v and t_0 (Ratcliff and Tuerlinckx, 2002). This three-fold integration with the infinite sum at its core is the reason why parameter estimation for the diffusion model becomes so expensive when computation of F_+ and F_- is slow.[2]

We will illustrate that efficient parameter estimation algorithms can be based on numerical solution of the PDE (1) instead of considering the infinite sum in eq. (2). A parameter estimation procedure that is based on such a numerical solution can be faster than traditional algorithms for three reasons: firstly, the infinite sum no longer needs to be calculated. Secondly, the PDE approach returns the CDF for all values of z in one loop of calculations. The best starting point z can be easily picked out without including this parameter in the costly multidimensional

[2] Of course an implementation of (2) still has a lot of room for optimisations: for example some of the integrals might be tractable analytically and there are alternative closed form expressions for F_+ which converge faster than (2) for small values of t.

search. Therefore the search space is reduced by one dimension. The third advantage takes effect if the variability parameters are included in the model, because the PDE approach allows the inclusion of variability in starting point and non-decisional component with little extra cost;[3] only variability in the drift rate requires extra calculations.

5 Comparing the PDE Algorithm with the Infinite-Sum Method

Comparing speed and efficiency of the PDE approach with algorithms based on eq. (2) leads to some difficulties: in both cases small changes in the diffusion model software may cause large changes in the trade-off between accuracy and computation time. For the closed-form solution these changes concern the calculation accuracy of eq. (2) (ϵ, see Appendix B in Voss et al., 2004). For the PDE approach the discretisation step size in time (Δt) and space (Δz), see Appendix B, influence accuracy and speed of the parameter estimation. Additionally, for both methods, the accuracy chosen in the evaluation of the integrals for the parameter variability and the details in the implementation of the simplex search have a big impact.

Since, ultimately, only the quality of the result and the time used to obtain this result are of interest, we perform two numerical experiments to compare accuracy and execution time of the algorithms. In the first experiment we compare the performance for computing a set of CDFs both using an implementation of the PDE method (*fast-dm*, Voss and Voss, in press) and using a recent implementation of the IS method (*DMAT*, Vandekerkhove and Tuerlinckx, in press, 2007). Accuracy is measured by comparing the results with CDFs computed with significantly higher accuracy (which takes far too long to be done in every-day application of the method). In the second experiment, we compare the performance of the complete parameter estimation procedure. For this purpose, one hundred random samples for one parameter set are generated. Then, parameter values are recovered using implementations of the PDE algorithm and the IS algorithm. Both computational approaches are used with the KS optimisation criterion and with the χ^2 approach. For both experiments the same computer with a dual core Pentium 4 processor (2.8 GHz) was used.

Experiment 1. For this study, target CDFs for 15 different parameter sets were calculated with high precision[4]. For the "standard" parameter set the values $a = 2$, $z = 0.5\,a$, $v = 0$, $t_0 = 0.2$, $s_z = 0$, $s_v = 0$, and $s_{t_0} = 0$ are used. In each of the remaining 14 parameter sets one of the values is modified (see Table 1 for details). For each of the resulting parameter sets, CDFs were calculated for both response alternatives from 0 ms to 5000 ms in steps of 10 ms. The results were merged into one combined CDF for each parameter set by mirroring the "error" distribution (cf. Voss et al. 2004 for details on this procedure).

In a second step all CDFs were calculated again using the *plot-cdf* tool of *fast-dm* (Voss and Voss, in press) with precision 3.0 (default) and precision 2.0 and using the command *cdfdif* of *DMAT* (Vandekerkhove and Tuerlinckx, in press) respectively. Accuracy was computed as the maximum absolute vertical distance between the newly computed CDFs and the corresponding target CDFs. Table 1 shows the computation time for this step and the number of correctly obtained decimal places in the result, that is, the negative base-10 logarithm of the accuracy values.

As can be seen, for most parameter sets the PDE method is noticeably faster while yielding similar or better accuracy. This is especially noteworthy because the PDE method provides results not only for the given value of z but for the whole possible range (from 0 to a) simultaneously. Accuracy of the IS method is especially problematic for large values of a: In this case, the infinite sum has to be evaluated very often to get reasonable results and processing times would increase to unacceptable values (which is prohibited in *DMAT* by a stopping criterion).

For large values of s_v the processing time of *fast-dm* increases strongly. However, this is

[3] To include inter-trial variability of the starting point it is necessary to aggregate data from models with different starting points; this information is automatically available in the numerical solution because CDFs are calculated for 'all' starting points from 0 to a simultaneously. Likewise, all CDFs for different values of t_0 are present, because changing t_0 only corresponds to a shift of the CDFs in t-direction.

[4] The CDFs that were used as reference to calculate the accuracy are available at *http://www.psychologie.uni-freiburg.de/Members/voss/fast-dm/materials*

Table 1: *Computation time (T) and accuracy (Acc) for 15 CDFs calculated both with the PDE method (fast-dm, Voss and Voss, in press) and the IS method (DMAT, Vandekerkhove and Tuerlinckx, in press). For the standard CDF the parameter values $a = 2$, $z = 0.5\,a$, $v = 0$, $t_0 = 0.2$, $s_z = 0$, $s_v = 0$, and $s_{t_0} = 0$ were used. For each of the remaining CDFs the denoted values are changed. Acc is calculated as $-log_{10}(T_{\max})$, where $T_{\max}$ is the maximal vertical distance from an highly accurate CDF.*

	PDE ($prec = 3.0^a$)		PDE ($prec = 2.0^a$)		IS ($DMAT^b$)	
Parameter Set	T [ms]	*Acc*	T [ms]	*Acc*	T [ms]	*Acc*
Standard	6	4.5	3	3.7	201	2.8
$a = 0.50$	2	3.5	1	2.6	103	5.2
$a = 4.00$	10	5.3	4	4.3	322	0.8
$z = 0.10\,a$	6	3.1	2	2.3	214	3.3
$z = 0.25\,a$	6	4.1	3	3.1	215	2.9
$v = 1.00$	6	4.2	3	3.4	234	4.4
$v = 4.00$	6	3.1	3	2.8	168	5.6
$t_0 = 0.05$	6	4.5	3	3.7	207	2.8
$t_0 = 1.00$	5	4.5	2	3.7	178	2.3
$s_z = 0.20\,a$	11	4.6	3	3.5	255	2.8
$s_z = 0.90\,a$	9	4.1	3	3.2	255	2.9
$s_v = 0.50$	178	3.4	10	2.4	236	2.9
$s_v = 3.00$	1082	3.5	53	2.6	235	1.9
$s_{t_0} = 0.50\,t_0$	30	3.2	2	1.6	261	2.8
$s_{t_0} = 2.00\,t_0$	56	3.2	3	2.1	263	2.8

Notes: [a] The component "plot-cdf" of *fast-dm-26* was used. [b] The procedure "cdfdif" of *DMAT* was used.

necessary to get the same precision as for the other parameter sets. As can be seen, the accuracy of *DMAT* is comparably poor here as well (this is caused by the fact that *DMAT* uses a constant number of steps for the calculation of the the corresponding integral).

In either computational approach, accuracy and processing time depend crucially on details of the implementation. In our implementation of the PDE approach, the precision value determines step size in discretising the PDE in "space" and time. The default value of 3.0 used in our software is chosen so that it guarantees an accuracy that is certainly good enough for all psychological applications. Whenever calculation speed is of importance the user can decide to decrease precision.

Experiment 2. The main aim of Experiment 1 is to demonstrate the accuracy of the PDE algorithm in the calculation of a complete CDF. With Experiment 2, the efficiency of the PDE algorithm in a parameter estimation procedure is analysed. Specifically, the performance of the PDE method and the infinite-sum method are compared using the Kolmogorov-Smirnov criterion or the χ^2 statistic. There are two advantages of the PDE method in the multi-dimensional parameter search: Firstly, the search space is reduced by one dimension as explained above. This reduction of the search space makes the simplex algorithm more efficient and more stable.

Perhaps even more important is the time dimension of the CDF: Using the PDE algorithm, the complete CDF is returned (with a given step-size Δt) in one cycle of calculations. Therefore, the estimation of the fit between the predicted and the empirical CDFs is—almost—independent of the number of data points used for the calculation of the optimisation criterion: Consequently, the duration of the parameter estimation should be similar for χ^2 and for KS, although for the χ^2 statistic the predicted CDF has to be evaluated only at the borders of each response time bin (e.g., 10 evaluations), while the the KS-statistic requires evaluations at each empirical response time (e.g., 200 evaluations).

To illustrate this, 100 random sets of 200 responses were simulated from the "standard" parameter sets from Table 1. The resulting distributions were used to recover parameter values with different estimation procedures: The PDE algorithm and the IS algorithm were used in a parameter search that was based on either the KS-statistic or on the χ^2 statistic as optimisation criterion. For the condition "PDE-KS" *fast-dm-26* (Voss and Voss, in press) with precision 2.0 was used. For the other conditions, *fast-dm* was adapted accordingly: For the calculation of χ^2 bins were chosen as done by Ratcliff (e.g., Ratcliff and Tuerlinckx, 2002) and for the calculation

Table 2: *Recoverd parameter values and processing times (means and SDs) from implementations based on different algorithms for the calculation of predicted CDFs and based on different optimisation criteria (Experiment 2). Parameters were recovered from 100 simulated data sets with 200 responses each (original values:* $a = 2.0$, $z = 0.5\,a$, $v = 0.0$, $t_0 = 0.2$, $s_z = 0.0$, $s_v = 0.0$, *and* $s_{t_0} = 0.0$*).*

Software	CDF calc.	Opt. crit.	a	z	v	t_0	s_z	s_v	s_{t_0}	$T[s]$
fast-dm[a]	PDE	KS	2.01	1.01	−0.00	0.25	0.64	0.29	0.25	4.51
			(0.08)	(0.09)	(0.10)	(0.03)	(0.11)	(0.12)	(0.07)	(1.08)
	PDE	χ^2	2.03	1.01	0.00	0.27	0.66	0.29	0.24	5.41
			(0.13)	(0.09)	(0.11)	(0.05)	(0.36)	(0.25)	(0.09)	(1.60)
	IS	KS	2.02	1.01	0.00	0.24	0.62	0.29	0.25	10.79
			(0.09)	(0.11)	(0.11)	(0.05)	(0.19)	(0.19)	(0.18)	(12.14)
	IS	χ^2	2.24	1.12	0.00	0.32	1.12	0.75	0.28	1.47
			(0.28)	(0.16)	(0.17)	(0.11)	(0.87)	(0.71)	(0.20)	(0.99)
DMAT	IS	χ^2	3.07	1.53	−0.01	0.31	1.69	2.09	0.16	230.32[b]
			(0.97)	(0.50)	(0.37)	(0.11)	(1.06)	(1.53)	(0.20)	(565.57)

Notes: PDE = Partial Differential Equation approach; IS = Infinite Sum approach; KS = Kolmogorov-Smirnov statistic.

[a] Only the values from the first data row (PDE/KS) were calculated with the published version of *fast-dm-26* (precision=2.0). For the following rows, adapted versions of *fast-dm* were used.

[b] The long mean calculation time is based—partially—on some outliers ($T_{\max} = 4171 sec.$). The median of the computation time was 24 seconds.

of the infinite sum, a copy of the underlying C-code (cdfdiff.c[5]) of *DMAT* (Vandekerkhove and Tuerlinckx, 2007) was incorporated into *fast-dm*.

Additionally, the recovery procedure was repeated with *DMAT* (Vandekerkhove and Tuerlinckx, 2007) as well. However, since other details of the parameter search (e.g., starting values for the SIMPLEX search, termination criterion of the SIMPLEX search) are not identical between *DMAT* and *fast-dm*, it is problematic to interpret the results as a comparison of calculation algorithms and optimisation criteria.

Table 2 shows the results of experiment 2.[6] For the PDE approach the calculation time does not vary much between KS and χ^2. This is so because—independently of the optimisation criterion—the complete CDF is calculated in each step of the SIMPLEX search. Accuracy of the recovered parameters is also similar.

For the IS method, the duration of the estimation procedure is linked directly to the required number of evaluations. Consequently, the IS method is slow when it is used in combination with KS, and fast when used with χ^2. Interestingly, the combination of IS and χ^2 leads to poorer results. Evidently, inaccuracies of both algorithms accumulate in an unfortunate way.

As mentioned above, the comparison with *DMAT* (Vandekerkhove and Tuerlinckx, 2007) has limited explanatory power because the programs are very different in many aspects. Nonetheless, the great differences between the IS/χ^2 version of *fast-dm* and *DMAT* are surprising, since the evaluations of the CDFs are based on the same code. There are several potential explanations of the long processing times of *DMAT*: Firstly, there was small a number of very big values in the calculation times. This was the case when the initial guess (i.e., the starting values for the SIMPLEX search) of *DMAT* fitted badly. Secondly, the simplex was run more often by *DMAT* (up to eight times when results were considered to be 'suspect') than by *fast-dm* (always three times). Third, the execution of C-code from the MATLAB environment may be slower compared to a 'pure' C-coded program. It remains unclear as well, why the results obtained from *DMAT* are less accurate than the results of *fast-dm* in the IS/χ^2 version.

6 Summary and Conclusions

In recent years the interest in diffusion model analysis has notably increased, an effect which is undoubtedly related to the increasing computational power that is nowadays available—and

[5] We thank Joachim Vandekerkhove and Francis Tuerlinckx for providing the source code.

[6] The data sets, the control file for fast-dm (experiment.ctl), and the command file used for the *DMAT* approach (DMAT.m) can be downloaded from
`http://www.psychologie.uni-freiburg.de/Members/voss/fast-dm/materials`

affordable—for every researcher. Nonetheless, estimating the parameters of the complete diffusion model for large numbers of participants and large numbers of trials is still a challenge for a normal office PC.

In the present paper we introduce a new method, mathematically equivalent to the "classical" closed-form solution of eq. (2), to calculate the predicted RT-distributions of the diffusion model. This new algorithm is based on the numerical solution of the PDE describing the diffusion model. The numerical PDE approach requires slightly more mathematical understanding; however, implementing it is worth the effort because parameter estimations that used to take hours can be completed in only a few minutes with this approach. We applied the proposed approach to the Kolmogorov-Smirnov method introduced by Voss et al. (2004), and to a χ^2-based parameter estimation. The PDE approach is especially apt for KS optimisation, because in this case many values from the same CDF are needed. These values can be provided very fast once the PDE has been solved. Nonetheless, the PDE methode can be implemented for the more widely-used χ^2 criterion as well. It is not in the scope of the present paper to compare the efficiency of KS and χ^2 approaches. However, it can be expected that KS is superior especially for large samples, while no substantial differences are to be expected for large samples. The efficency of the PDE method may prove useful to facilitate and speed up further research in this area.

A The Kolmogorov Backward Equation

Since the partial differential equations for the distribution functions F_+ and F_- form the core of the proposed method, and since the Kolmogorov backward equation in the form required for our results is not easily found in the literature, we present here a complete proof of the fact that F_+ can be obtained as the solution of the given PDE.

We state the result for slightly more general diffusion processes than the Brownian motion with constant drift considered in the previous sections. In our application we apply the theorem for the constant drift function $v(x) = v$.

Theorem. Let Z be a solution of the stochastic differential equation (SDE)

$$dZ_t = v(Z_t)\,dt + dB_t, \quad Z_0 = z$$

where the drift $v\colon \mathbb{R} \to \mathbb{R}$ is C^∞ with bounded first derivative, $0 \leq z \leq a$, and B is a standard Brownian motion. Let $F_+(t, z)$ be the probability that Z hits a before time t and before it hits 0. Then F_+ satisfies the PDE

$$\frac{\partial}{\partial t} F_+(t,z) = \frac{1}{2}\frac{\partial^2}{\partial z^2} F_+(t,z) + v\frac{\partial}{\partial z} F_+(t,z) \quad \text{for all } t > 0,\ 0 < z < a$$

with boundary conditions

$$F_+(t,0) = 0, \quad F_+(t,a) = 1 \qquad \text{for all } t > 0$$

and initial condition

$$F_+(0,z) = \begin{cases} 0 & \text{if } 0 \leq z < a \text{ and} \\ 1 & \text{if } z = a. \end{cases}$$

Proof. Let $0 < t < T$ and $D = (0,T) \times (0,a) \subseteq \mathbb{R}^2$. Define the two-dimensional process X by $X_s = (s, Z_s)$ for all $s \geq 0$. Then X solves the SDE

$$dX_s = \begin{pmatrix} 1 \\ v(Z_s) \end{pmatrix} ds + \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} d\tilde{B}_s$$

where $\tilde{B}$ is a two-dimensional standard Brownian motion. Furthermore, let $S = \inf\{\, s \geq 0 \mid X_s \notin D \,\}$ be the first exit time of X from the rectangle D, define a function f on the boundary of D by $f(s,z) = 1$ if $z = a$ and $f(s,z) = 0$ else, and let

$$\phi(s,z) = \mathbb{E}_{s,z}\big(f(X_S)\big) \quad \forall (s,z) \in [0,T] \times [0,a],$$

where $\mathbb{E}_{s,z}$ denotes the expectation for the process X starting at $X_0 = (s, z)$. Since $f(X_S) = 1$ if and only if $Z_S = a$, we have

$$F_+(t, z) = \phi(T - t, z). \tag{3}$$

From Example 7.5.6 of Øksendal (1998) we know that ϕ satisfies

$$\mathcal{A}\phi(s, z) = 0 \quad \forall (s, z) \in D \tag{4}$$

where the generator $\mathcal{A}$ is given by Theorem 7.5.4 in Øksendal (1998):

$$\mathcal{A}\phi(s, z) = 1 \frac{\partial}{\partial s}\phi(s, z) + v(z)\frac{\partial}{\partial z}\phi(s, z) + \frac{1}{2}\frac{\partial^2}{\partial z^2}\phi(s, z).$$

Here we used the fact that by elliptic regularisation (e.g., Hörmander, 1983, Theorem 13.4.1) the solution of (4) is C^∞ on D. By solving this relation for $\frac{\partial}{\partial z}\phi(s, z)$ we find

$$\begin{aligned}\frac{\partial}{\partial t}F_+(t, z) = -\frac{\partial}{\partial s}\phi(T - t, z) = v(z)\frac{\partial}{\partial z}\phi(T - t, z) + \frac{1}{2}\frac{\partial^2}{\partial z^2}\phi(T - t, z) \\ = v(z)\frac{\partial}{\partial z}F_+(t, z) + \frac{1}{2}\frac{\partial^2}{\partial z^2}F_+(t, z).\end{aligned}$$

This is the required PDE. The fact that F_+ satisfies the given boundary conditions and initial conditions can be read off from the definition of ϕ using the relation (3). ∎

The corresponding PDE for F_-, the probability that Z hits 0 before time t and before it hits a, can be obtained either by choosing $f(s, z) = 1$ if $z = 0$ and $f(s, z) = 0$ else in the proof, or by the reflection argument presented in section 4.

B Numerical Solution of PDEs

The algortihm described in this article involves numerical solution of the parabolic PDE (1). Solving PDEs numerically is a well-established topic both in mathematics and in applied areas. There exists a vast body of literature and there are also many ‘black box’ PDE solvers available (e.g. the PDE toolbox of Matlab). Nevertheless, in order to keep the text as self-contained as possible, we give here a rough sketch of the numerical method used in our own implementation “fast-dm” (Voss and Voss, in press). For a more detailed description we refer to the following text books: the theoretical background is described in Boyce and DiPrima (2001) for ordinary differential equations and Strauss (1992) for PDEs. How to solve these equations numerically is, for example, described in Iserles (1996) for ordinary differential equations and Morton and Mayers (1994) for PDEs. A short and very accessible exposition can also be found in Press, Teukolsky, and Vetterling (1992, chapter 19).

The key to the numerical solution of PDEs such as eq. (1) by using finite difference methods is to discretise ‘space’ z and time t: instead of the full z-interval $[0, a]$ we only consider the discrete set of z-values $\{0, \Delta z, 2\Delta z, \ldots, N\Delta z\}$ where $\Delta z = a/N$ for some positive integer N and instead of the t-interval $[0, \infty)$ we only consider the t-values $\{0, \Delta t, 2\Delta t, \ldots\}$. The algorithm computes values $F_{i,j}$ which approximate the true solution F_+ by

$$F_{ij} \approx F_+(i\Delta t, j\Delta z) \quad \text{for } i = 0, 1, 2, \ldots \text{ and } j = 0, 1, \ldots, N.$$

The accuracy of this approximation depends on the step sizes Δt and Δz. The smaller these values are, the more accurate is the approximation (and the more expensive is the algorithm).

The algorithm works by considering a grid row with fixed t at a time, starting with an approximation of the initial condition (1c):

$$(F_{00}, \ldots, F_{0N}) = (0, \ldots, 0, 1).$$

Then, in each step, the algorithm uses the approximation $(F_{i0}, \ldots, F_{iN})$ for time $i\Delta t$ to compute an approximation for time $(i + 1)\Delta t$.

To simplify the presentation we consider the function

$$u(t,z) = F_+(t,z) - \frac{1-\exp(-2vz)}{1-\exp(-2va)}$$

instead of F_+ for the remaining part of the section. A simple calculation shows that u solves that same PDE as F_+, but with homogeneous boundary conditions $u(t,0) = u(t,a) = 0$ instead of (1b). Again, we denote the computed solution for time $n\Delta t$ by $u^n = (u_1^n, \ldots, u_{N-1}^n)$. We do not include the outermost points u_0^n and u_N^n since these are always zero due to the boundary conditions.

The partial derivatives present in the PDE are approximated by the following finite differences:

$$\frac{\partial}{\partial t}u \approx \frac{u(t+\Delta t, z) - u(t,z)}{\Delta t}, \quad \frac{\partial}{\partial z}u \approx \frac{u(t, z+\Delta z) - u(t, z-\Delta z)}{2\Delta z},$$
$$\frac{\partial^2}{\partial z^2}u \approx \frac{u(t, z+\Delta z) - 2u(t,z) + u(t, z-\Delta z)}{\Delta z^2}.$$

Using these approximations we can then write the application of the differential operator

$$\mathcal{L} = \frac{1}{2}\frac{\partial^2}{\partial z^2} + v\frac{\partial}{\partial z} \tag{5}$$

as a matrix vector multiplication: collecting all the terms we get

$$\mathcal{L}u(n\Delta t, \,\cdot\,) \approx L^N u^n$$

where L^N is the tri-diagonal matrix given by

$$L^N = \begin{pmatrix} -\frac{2}{2\Delta z^2} & \frac{1}{2\Delta z^2} + \frac{v}{2\Delta z} & \\ \frac{1}{2\Delta z^2} - \frac{v}{2\Delta z} & -\frac{2}{2\Delta z^2} & \frac{1}{2\Delta z^2} + \frac{v}{2\Delta z} \\ & \frac{1}{2\Delta z^2} - \frac{v}{2\Delta z} & -\frac{2}{2\Delta z^2} \end{pmatrix} \in \mathbb{R}^{(N-1)\times(N-1)}.$$

The middle row in this matrix is repeated along the diagonal $N-3$ times in order to get the full matrix.

The approximations introduced above suggest the following approximation to the PDE:

$$\frac{u^{n+1} - u^n}{\Delta t} = L^N\bigl(\theta u^{n+1} + (1-\theta)u^n\bigr) \tag{6}$$

where $\theta \in [0,1]$ is a parameter of the method. For $\theta = 0$ the derivative on the right hand side is evaluated only for the current approximation u^n. For all values $\theta > 0$ the derivative is evaluated for a mixture of u^n and u^{n+1}. In these cases one has to solve a system of linear equations to compute u^{n+1} from u^n: by rearranging the terms in (6) we get

$$\bigl(I - \Delta t\theta L^N\bigr)u^{n+1} = \bigl(I + \Delta t(1-\theta)L^N\bigr)u^n \tag{7}$$

where I is the $(N-1)\times(N-1)$ identity matrix. As we will discuss below, the choice of the parameter θ affects the stability of the method. Common choices are $\theta = 0$ (Euler scheme), $\theta = 1/2$ (Crank Nicolson scheme) and $\theta = 1$ (implicit Euler scheme). In our own implementation we use the Crank Nicolson method.

In order to understand stability and convergence of the resulting method, it is useful to consider the Fourier transform of the solution: one can check that the differential operator $\mathcal{L}$ from (5) has eigenvalues $\lambda_k = -\frac{1}{2}(\pi^2k^2 + v^2)$ and that the corresponding eigenfunctions $f_k(z) = \sin(\pi kz)\mathrm{e}^{-vz}$ for $k = 1, 2, \ldots$ form a basis of the space of square integrable functions. It transpires that the eigenvalues $\lambda_1^N, \ldots, \lambda_{N-1}^N$ of the approximation L^N are still strictly negative and that the corresponding eigenvectors $v_1^N, \ldots, v_{N-1}^N$ form a basis of $\mathbb{R}^{N-1}$. We express the approximations u^n in this basis as

$$u^n = \sum_{i=1}^{N-1} \alpha_i^n v_i^N.$$

Substituting this into (7) and using the relation $L^N v_i^N = \lambda_i^N v_i^N$, we find the the time evolution of the coefficients α_i^n is given by

$$\alpha_i^{n+1} = \frac{1 + \Delta t(1-\theta)\lambda_i^N}{1 - \Delta t\theta\lambda_i^N}\,\alpha_i^n =: q_i\alpha_i^n \tag{8}$$

for all $i = 1, \ldots, N-1$, $n \in \mathbb{N}$, where the α_i^0 are found from the intial condition.

The method is stable if all coefficients q_i in (8) satisfy $|q_i| < 1$ (for $|q_i| > 1$ the coefficients α_i^n grow exponentially). Using the fact that the λ_i^N are negative, we find that $|q_i| < 1$ is equivalent to the condition

$$\Delta t(1-2\theta) < -\frac{2}{\lambda_i^N} \quad \forall i = 1, \ldots, N-1,$$

where the right hand side is positive. For $\theta \geq 1/2$ this relation is always satisfied and thus the method is stable, independently of the choice of Δt. For $\theta < 1/2$ one gets a bound on Δt, depending on the smallest of the λ_i^N. A more detailed analysis shows that this eigenvalue is approximately equal to $-4/\Delta z^2$ (with exact equality for $v = 0$) and thus for $\theta < 1/2$ the method is stable only if $\Delta t < \frac{1}{2(1-2\theta)}\Delta z^2$.

Convergence of the method to the correct result must be proved separatly from stability. It is a consequence of the fact that the biggest eigenvalue of L^N is bounded away from 0, uniformly in N. Here we restrict discussion to the following observation: since we subtracted the stable solution of the PDE when switching from F_+ to u, the exact solution u converges to 0 as $t \to \infty$. To get an approximation for the solution at time $t = n\Delta t > 0$ we need to perform $t/\Delta t$ steps of the discretised method. Using the approximation $q_i = 1 + \Delta t\lambda_i^N + O(\Delta t^2)$, the coefficients are then

$$\alpha_i^n \approx (1 + \Delta t\lambda_i^N)^{t/\Delta t} \to \exp(\lambda_i^N t)$$

as $\Delta t \downarrow 0$ and thus the solution of the numerical scheme also converges to 0 as $t \to \infty$. This shows that the discrete solution converges to the exact result at least asymptotically for $t \to \infty$. A similar argument can be used to show convergence of the method for every fixed t.

We conclude by emphasising that the Crank-Nicolson method used in our implementation is only one of many possible methods. For example, since the eigenvalues and eigenfunctions of the operator $\mathcal{L}$ are explicitly known, one could use spectral methods. These methods are closer in spirit to the infinite sum method.

References

Boyce, W. E., DiPrima, R. C., 2001. Elementary Differential Equations and Boundary Value Problems, seventh Edition. John Wiley & Sons.

Feller, W., 1971. An Introduction to Probability Theory and its Applications, 2nd Edition. Vol. II. John Wiley & Sons.

Grasman, J., van Herwaarden, O. A., 1999. Asymptotic Methods for the Fokker-Planck Equation and the Exit Problem in Applications. Springer.

Hörmander, L., 1983. The Analysis of Linear Partial Differential Operators II. Springer.

Iserles, A., 1996. A First Course in the Numerical Analysis of Differential Equations. Cambridge University Press.

Morton, K., Mayers, D., 1994. Numerical Solution of Partial Differential Equations. Cambridge University Press.

Nelder, J., Mead, R., 1965. A simplex method for function minimization. Computer Journal 7, 308–313.

Øksendal, B. K., 1998. Stochastic Differential Equations: An Introduction With Applications. Springer.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., 1992. Numerical Recipes in C, 2nd Edition. Cambridge University Press.

Ratcliff, R., 1978. A theory of memory retrieval. Psychological Review 85 (2), 59–108.

Ratcliff, R., 2002. A diffusion model account of response time and accuracy in a brightness discrimination task: Fitting real data and failing to fit fake but plausible data. Psychonomic Bulletin and Review 9 (2), 278–291.

Ratcliff, R., Rouder, J. N., 1998. Modeling response times for two-choice decisions. Psychological Science 9 (5), 347–356.

Ratcliff, R., Spieler, D., McKoon, G., 2000. Explicitly modeling the effects of aging on response time. Psychonomic Bulletin and Review 7 (1), 1–25.

Ratcliff, R., Thapar, A., Gomez, P., McKoon, G., 2004. A diffusion model analysis of the effects of aging in the lexical-decision task. Psychology & Aging 19 (2), 278–289.

Ratcliff, R., Thapar, A., McKoon, G., 2001. The effects of aging on reaction time in a signal detection task. Psychology & Aging 16 (2), 323–341.

Ratcliff, R., Thapar, A., McKoon, G., 2003. A diffusion model analysis of the effects of aging on brightness discrimination. Perception & Psychophysics 65 (4), 523–535.

Ratcliff, R., Tuerlinckx, F., 2002. Estimating parameters of the diffusion model: Approaches to dealing with contaminant reaction times and parameter variability. Psychonomic Bulletin & Review 9 (3), 438–481.

Ratcliff, R., Van Zandt, T., McKoon, G., 1999. Connectionist and diffusion models of reaction time. Psychological Review 106 (2), 261–300.

Spaniol, J., Madden, D. J., Voss, A., 2006. A diffusion model analysis of adult age differences in episodic and semantic long-term memory retrieval. Journal of Experimental Psychology: Learning, Memory, and Cognition 32 (1), 101–117.

Strauss, W. A., 1992. Partial Differential Equations, An Introduction. John Wiley & Sons.

Vandekerkhove, J., Tuerlinckx, F., 2007. Fitting the ratcliff diffusion model to experimental data. Manuscript submitted for publication.

Vandekerkhove, J., Tuerlinckx, F., in press. Diffusion model analysis with matlab: A dmat primer. Behaviour Reserch Methods.

Voss, A., Rothermund, K., Brandtstädter, J., in press. Interpreting ambiguous stimuli: Separating perceptual and judgmental biases. Journal of Experimental Social Psycholgy.

Voss, A., Rothermund, K., Voss, J., 2004. Interpreting the parameters of the diffusion model: An empirical validation. Memory & Cognition 32, 1206–1220.

Voss, A., Voss, J., in press. Fast-dm: A free program for efficient diffusion model analysis. Behaviour Reserch Methods.

Wagenmakers, E.-J., van der Maas, H. L. J., Grasman, R. P. P. P., 2007. An ez-diffusion model for response time and accuracy. Psychonomic Bulletin & Review 4, 3–22.