Residue iteration decomposition (RIDE): A new method to separate ERP components on the basis of latency variability in single trials

GUANG OUYANG, GRIT HERZMANN, CHANGSONG ZHOU, and WERNER SOMMER

Abstract

Event-related brain potentials (ERPs) are important research tools because they provide insights into mental processing at high temporal resolution. Their usefulness, however, is limited by the need to average over a large number of trials, sacrificing information about the trial-by-trial variability of latencies or amplitudes of specific ERP components. Here we propose a novel method based on an iteration strategy of the residues of averaged ERPs (RIDE) to separate latency-variable component clusters. The separated component clusters can then serve as templates to estimate latencies in single trials with high precision. By applying RIDE to data from a face-priming experiment, we separate priming effects and show that they are robust against latency shifts and within-condition variability. RIDE is useful for a variety of data sets that show different degrees of variability and temporal overlap between ERP components.

Descriptors: mental chronometry, event-related potentials, single-trial responses, component separation, face recognition

Mental Chronometry and ERP-Latency Variability

Information processing in biological systems, that is, cognition or thinking in higher organisms, is time-consuming and proceeds in separate stages or levels. Mental chronometry seeks to identify these subprocesses of cognition and their timing (Posner, 2005). Using the electrical or magnetic signals of the brain (electroencephalogram [EEG] and magnetoencephalogram [MEG]) in mental chronometry is a very attractive approach because these signals provide direct insight into the mental subprocesses and their timing. In contrast, behavioral methods can only inform about the end products of information processing.

Despite the significant advances made by applying EEG and MEG to cognitive neuroscientific questions (e.g., Rugg & Coles, 1995), some of the biggest challenges lie in the poor signal-to-noise ratio and in unambiguously interpreting event-related potentials (ERPs; and event-related electromagnetic fields). The averaging method, employed in ERP research, uses events like stimuli or responses to synchronize groups of EEG epochs. These groups of typically 30 to 60 trials belonging to the same experimental condition are then averaged time point by time point, yielding complex wave shapes, which consist of separate components. At least some of these components presumably reflect specific cognitive subprocesses. If the psychological significance of a specific ERP component is known, differences in amplitudes and latencies between experimental conditions can be used to infer the underlying functional differences between these conditions. For example, the lateralized readiness potential (LRP) in the ERP has been related to response-hand specific motor activation (Coles, 1989); therefore the interval between a stimulus and the LRP onset can be used to measure the time demands for perceptual processes and response selection processes, whereas the interval between the LRP onset and the response reflects the duration of motor processes (Masaki, Wild-Wall, Sangals, & Sommer, 2004; Osman, Moore, & Ulrich, 1995). Unfortunately, the situation is less clear for other components, for example, the P300. This component is taken to reflect stimulus classification because it is sensitive to the task-defined category of preceding processes but not on the following processes, such as motor programming and execution (Kutas, McCarthy, & Donchin, 1977); however, this position is contested (Verleger,
Many years ago it has been recognized that amplitudes and latencies of ERP components vary from trial to trial within a given experimental condition. When such intrinsically variable signals are averaged, it leads to an overlap between adjacent components and to smearing of the time-variable components, broadening their shape and diminishing their amplitude. Therefore, the interpretation of ERP latencies and their relationships to cognitive processing is a challenge in mental chronometry. More specifically, one difficulty in relating cognitive stages to averaged ERPs lies in the ambiguity of ERP latencies, which largely results from the following two reasons.

1. Trial-to-trial variability of component latencies. Previous experiments have shown, for example, that the latency of the P300 or P3 (also known as late positive complex [LPC]) is positively correlated with reaction time (RT), indicating that the variation of RT is partially due to the fluctuation of the time demands of stimulus classification (Kutas et al., 1977; McCarthy & Donchin, 1981; Polich, 2007). These latency fluctuations between trials result in a smearing effect on the shape of the LPC in the averaged ERP. This leads to distortions of (peak) latencies, reductions in maximum (peak) amplitudes, and a broadening of the component, advancing its onset and delaying the offset. If latency variability differs between conditions or participant groups, it also induces a bias into the interpretation of ERP differences in terms of amplitude changes, latency variation, or both.

2. Overlap of different ERP components. ERP components may consist of separate subcomponents, which are more or less related to response speed, showing variable or stationary latency with respect to RT, as has been shown for the P300 or late positive complex (e.g., Hohnsbein, Falkenstein, Hoormann, & Blanke, 1991; for review, see Verleger, 1997). Thus the measurement of latency-variable components may be affected by and overlap with latency-invariant components. Consequently, the latency detected from a composite ERP component may be ambiguous also with respect to the timing of the (multiple) underlying cognitive stages.

Taken together, conventional ERP averaging is insufficient to reveal the subtle changes in cognitive activity related to task processing. Therefore, a number of alternative methods have been suggested, such as locking averaged ERP to various non-stimulus events, Woody filtering, ERP deconvolution, independent component analysis, and RT binning, all aiming to solve this problem from different perspectives and with different requirements. In the following, we will briefly review these methods.

**Brief Review of Previous Work**

Stimulus-locked, electromyogram (EMG) onset-locked, key press onset-locked ERPs, or other kinds of event-synchronized ERPs (Handy, 2004) are a straightforward way to improve the accuracy of the timing and amplitude of ERP components with respect to a particular event. This method is easy to use and widely applied in many situations, such as the study of stimulus–response compatibility (e.g., Masaki et al., 2004). Still, as described above, if ERP components in the processing chain become uncoupled in time from the event used for synchronization, the average will be blurred and distorted.

Woody (1967) suggested a method, which seeks to detect the latencies of components such as the P3 in single-trial ERPs, based on an iterative strategy: First, latencies are estimated from the cross-correlation between the average ERP as a first template and each single-trial waveform. Then, all single-trial waveforms are averaged, aligning them to the estimated latency, thereby obtaining a new template. Finally, these steps are repeated until the successive templates converge. Woody’s method relies on the prerequisite that the component in question is monolithic, varying in latency across trials essentially without shape distortion. However, in reality, many parts of the ERP may not be monolithic components. As mentioned above, mixtures of latency-invariant and latency-variable components may bring about systematic errors in detecting latencies of interest. For this reason, other latency detection methods, such as peak-detection (Jaśkowski & Verleger, 1999, 2000), face the same limitations as Woody’s cross-correlation method.

Other methods to separate ERP components, such as principle component analysis (PCA; since Streeter & Raviv, 1966; Donchin, 1966) or independent component analysis (ICA; Jung et al., 2001), make strong assumptions and have difficulties in the interpretation of the results. ICA assumes that there are independent sources generating signals, which are projected onto the brain scalp. A number of components, up to the number of electrodes, will be obtained based on this assumption. However, most of them are not easy to interpret in terms of psychophysiological meaning. Especially, ICA-derived components are very likely to overlap with each other in time, rendering segregation according to latencies problematic. PCA mainly aims to separate a few leading components that are orthogonal to each other and contribute mostly to the variance in the record. The weak point is that PCA only assumes that the amplitude varies across trials without considering the latency (Möcks & Verleger, 1986). Therefore, the components obtained by ICA or PCA may not be optimal for temporal segmentation of ERP components.

Reaction time binning (Poli, Cinel, Citi, & Sepulveda, 2010) assumes that averaging trials according to RT bins can improve the power of ERP averages and solve the problem of identifying latency-variable ERPs and measuring their latencies. The underlying assumption is that for the epochs within a given RT bin, stimulus-locked, response-locked, as well as other fixed and latency-variable ERP components are much more synchronized than across bins. This assumption, however, is most likely invalid when the RT contains significant contributions to various subprocesses, like stimulus evaluation, response selection, and motor action, because their relative latencies can vary strongly even for trials with very similar RTs. The method has additional difficulties in determining the optimal bin size because of the usually skewed distribution of RTs. It improves only for situations with long RTs but cannot find component latencies for single trials.

There are also more direct methods that deal with the described ERP limitations. For example, Gerson, Parra, and Sajda (2005) divided EEG trials into short intervals from stimulus to response onset. They then used a weighted summation of single trials across the scalp to optimally discriminate activity between target and distractor conditions, after which they checked its contribution to response variability. Verleger, Jaśkowski, and Wascher (2005) used a simple method to examine the role of the P3b in task processing by averaging quartiles of stimulus- and response-synchronized EEG trials and compared the latency change between those two groups. They found that the P3b is absolutely locked neither to the stimulus nor to the
RIDE: A new method to separate ERP components

response and concluded that the P3b connects stimulus- and response-related processes.

An alternative class of methods, based on deconvolution, attempts to overcome the problems of event-locked, average ERPs. These methods consider that ERPs may consist of several components, which have at least partially independent latency variability and are correlated with different events. They attempt to separate stimulus-locked and response-locked ERP components assuming a linear model of ERP interaction. In particular, it is assumed that the recorded signal is the sum of at least two ERP components, one stimulus-locked $s(t)$ and one response-locked $r(t)$, with the shape of these ERPs being independent of RT. Under this assumption, it is possible to separate these ERP components using the distribution $\rho(t)$ of the RT because the stimulus-locked and response-locked averages $a_i(t)$ and $b_i(t)$ are related by linear convolution equations $a_i(t) = s(t) + r(t)\rho(t)$ and $b_i(t) = r(t) + s(t)\rho(t)$, which can be solved in Fourier space (Hansen, 1983; Takeda, Yamanaka, & Yamamoto, 2008; Zhang, 1998). Takeda, Yamanaka, and Yamamoto (2008) pointed out that this method will introduce artifacts into the resulting components by amplifying slow noise components ($\sim$ 1 Hz). A major limitation is that the method cannot work for latency-variable ERP components that are not strongly locked to the response. Such situations, however, may be quite typical in reality because of the contribution of various processes to RT in complex cognitive tasks. Obviously the deconvolution method will fail to find a latency-variable component if the time marker for this component is not available, for example, in no-go tasks or tasks where the response is not speeded (as in ratings or psychophysical judgments), not required in all trials, or complex (ratings, verbal reports).

Takeda et al. addressed the problem of identifying the unknown time markers by using an optimization scheme (Takeda, Sato, Yamanaka, Nozaki, & Yamamoto, 2010; Takeda, Yamanaka, Nozaki, & Yamamoto, 2008). For example, in no-go data, they estimated the latency of response-locked components by searching in all trials for a set of optimal latencies that minimize an objective function. Specifically, two components can be obtained: a Fourier transformation based on the stimulus onset and a set of random latencies, which are regarded as response times in single trials (Takeda, Yamanaka, & Yamamoto, 2008). After subtracting these components at the time markers, the summation of the remaining signals gives the objective function, which will be minimized if the latencies are correctly chosen and consequently the components are correctly recovered. This way, the search for ERP latencies becomes an optimization problem. However, the configuration space for a globally optimized solution is extremely large, rendering global search impractical. Takeda and coworkers therefore used a random search strategy, which, however, gets easily trapped within local minima, giving wrong results, especially in the presence of noise and when the number of ERP components increases.

Although deconvolution methods for separating latency variable components are appealing, they have not received much attention, possibly because, in addition to technical difficulties, there are few convincing demonstrations of their usefulness for improving the understanding of cognitive processes. For example, Takeda, Yamanaka, Nozaki, and Yamamoto (2008) reported that in a go/no-go experiment, the estimated RTs can show positive, negative, or no correlation with the real RTs in go data. In the present article, we introduce a new method for ERP decomposition. It is in line with the linear superposition model proposed by Hansen (1983), Takeda et al. (2010), Takeda, Yamanaka, Nozaki, and Yamamoto (2008), Takeda, Yamanaka, and Yamamoto (2008), and Zhang (1998), but overcomes many of the difficulties of the previous methods. In addition, we demonstrate the usefulness of the method for improving the interpretation of ERP findings from a face priming experiment.

Residue Iteration Decomposition

Improving mental chronometry with latency-variable ERP components must solve two basic problems: (a) separating ERP components associated with different mental processes and (b) measuring the latency of these components in single-trial EEG. Here we propose a method that can solve these two problems simultaneously based on the iterative use of the averaged ERP and the residues of single-trial ERPs. We call our method the residue iteration decomposition (RIDE).

In line with standard ERP phraseology, we consider ERPs to consist of components. Components can be distinguished by their polarity, latency range, scalp topography, and sensitivity to experimental conditions. Although ERP components may be variable in latency, some components, although distinguishable with regard to the parameters above and their functional significance, may be highly correlated in time. Such components appear as clusters and will be termed as such.

Method Description

RIDE originates from the basic idea that all the information about trial-to-trial variability is contained in their residues, that is, the differences between each single-trial EEG and the average ERP, which is the average of these single-trial epochs. For instance, the residues of stimulus-synchronized average ERP ought to display a closer resemblance to the latency-variable ERP components in those epochs than the raw, single-trial epochs, especially when the latency distribution of response-synchronized ERP components is broad. Thus, if we average the residues aligned to the correct latency or a good estimation of it, it is possible to restore the shape of the latency-variable ERP to some extent. On the basis of this estimated component, we can go back to modify the shape of the stationary (stimulus-locked) ERP and obtain new, improved single-trial residues and a better estimation of their latencies. Iteration of this procedure allows obtaining latency-variable ERPs and their latencies in single trials, together with the stimulus-locked ERPs.

RIDE assumes a linear superposition model of single-trial ERPs. First let us consider the simplest model, named the S-R model. Except for the noise $\xi$, the single-trial EEG is assumed to consist of two component clusters, labeled $S$ and $R$, which are locked to two different time markers, for example, the stimulus onset and the response, respectively. The term component cluster refers to all of the components whose latencies are locked to each other but could be related to several different sources. For example, those components with stable latency with respect to the stimulus onset will be classified into component cluster $S$. A single-trial EEG containing background noise can thus be expressed as

$$\text{EEG}_i(t) = S(t) + R(t + \tau_i) + \xi,$$

where $\tau_i$ is the latency of component cluster $R$ in the $i$th trial. In Figure 1, we present such a model schematically by taking two sine functions for the component clusters $S$ and $R$, in particular,
$S(t) = 15 \cdot \sin(\pi t/500)$ and $R(t) = 5 \cdot \sin(\pi t/200)$, disregarding noise for clarity’s sake. The latency $\tau_i$ follows a distribution $\rho(t)$, which is assumed to be Gaussian here; the method, however, is valid for all distributions. Figure 1A shows three single trials for the smallest, intermediate, and the largest latency of component cluster $R$. We can see that $R$ “rides” on the stationary component cluster $S$.

From Equation 1, we can see that the averaged ERP locked to $S$ (Figure 1a, black curve) can be described as

$$< ERP >_S = S(t) + \frac{1}{N} \sum_{i=1}^{N} R(t + \tau_i) + \frac{\xi}{\sqrt{N}}$$

$$= S(t) + \int R(t + \tau) \rho(t) dt + \frac{\xi}{\sqrt{N}}$$

$$= S(t) + R^n \rho + \frac{\xi}{\sqrt{N}}.$$  \tag{2}

Here $N$ is the number of total trials and $^n$ denotes the convolution operation. Equation 2 shows that simple averaging will smear the component cluster $S$ by the convolution of the latency-variable component cluster $R$, although noise is effectively reduced. On the other hand, the latency of the component cluster $R$ may not be estimated precisely from single-trial EEG. Let us now pay attention to the first trial. The actual latency $\tau_i$ (vertical dotted line “1” in Figure 1A) is different from the peak of the single-trial EEG (line “2” in Figure 1A). All methods detecting component latencies from single-trial EEG, such as simple peak detection or Woody’s method, will be systematically biased to larger values for short latencies and to smaller values for long latencies. These systematic errors become larger when the amplitude of the component cluster $R$ is small as compared to $S$.

Now let us consider the ERP residues in single trials:

$$Res_i = EEG_i - < ERP >_S = R(t + \tau_i) - R^n \rho.$$  \tag{3}

Figure 1. Model simulation of RIDE. Without considering noise, the single-trial EEG is a linear sum of a stationary component cluster $S(t) = 15 \cdot \sin(\pi t/500)$ and a latency-variable component cluster $R(t) = 5 \cdot \sin(\pi t/200)$ with a random distribution of latency $\tau_i$. Trial number: 100, Gaussian variance of latency: 50. A: Examples of three single trials, 1st (blue), 50th (red), and 100th (green), of simulated EEG epochs with the smallest, intermediate, and largest latency, respectively. The black line represents the average ERP. B: The corresponding residues of the single-trial ERPs. The vertical dashed lines “1,” “2,” and “3” indicate the real latency $\tau_i$ (peak) of the component cluster $R$, the peak of single-trial EEG, and the peak of the residue of the first trial, respectively. C: The convergence of the estimated component cluster $S$ to the real one during the iterative, purifying process of RIDE. D: The convergence of the component cluster $R$ to the real one in the 1st, 50th, and 100th trials. E: Mean relative error of the converged latency by RIDE in the parameter space of the latency distribution variance $\sigma_L$ and jitter distribution variance $\sigma_J$. F: The comparison of RIDE and random search method (Takeda et al., 2016; Takeda, Yamanaka, Nozaki, & Yamamoto, 2008) over 50 realizations with different reshuffled phase of the noise in each realization. The horizontal axis represents the noise level defined as the ratio of standard deviation of noise over that of the component $R$. The vertical axis represents the correlation coefficient between detected latency and given latency of $R$. 
For ease of discussion, we now drop the noise term. Figure 1B shows the three residues corresponding to the single trials in Figure 1A. We can see that each residue provides the information for the corresponding latency-variable component cluster R, although the shape is distorted. The peak of the residue (indicated by the dotted line “3” in Figure 1B) is close to the actual latency (line “1” in Figure 1A), but biased to smaller values for short latencies and to larger values for long latencies. Now if these residues are aligned to their latencies $\tau$ and averaged, all distortions are diminished, and we obtain a first estimate for R, which can be mathematically expressed as

$$R_1(t) = (\text{Res}) = R(t) - (R^*p)^*p;$$

here the term $(R^*p)^*p$ results from the fact that the term $R^*p$ in Equation 3 is now spread according to the distribution $p$ when the residue trials are aligned at latency $\tau$. With this estimate for component cluster R, we can go back to Equation 2 to obtain an improved estimate of component cluster S by replacing R with $R_1$, namely,

$$S_1 = < ERP >_S - R_1^*p = < ERP >_S - (R - (R^*p)^*p)^*p = < ERP >_S - R^*p - R^*p^*p^*p = S - R^*p^*p^*p^*p;$$

Now the procedure can be iterated if we use the estimated $S_1$ as ERP$_S$ and obtain improved residues Res = EEG – $S_1$. Iteration for $n$ steps, the two component clusters can be expressed as

$$S_n = S - R^*p^*p^*\ldots p \rightarrow S,$$

$$R_n = R - R^*p^*p^*\ldots p \rightarrow R$$

which will converge to S and R, respectively, because many iterative convolutions of the component cluster R, $R^*p^*p^*\ldots p$, will approach 0. Figures 1C and ID show the precise convergence of component clusters S and R to their real shapes, respectively. Unlike the method suggested by Takeda et al. (2010), Takeda, Yamanaka, Nozaki, and Yamamoto (2008), and Takeda, Yamanaka, and Yamamoto (2008), RIDE does not introduce any systematic artifacts, such as amplifying slow noise.

RIDE can be extended to the separation of more than two component clusters once their latencies are known. What we need to do is to align the residues to the latencies of the component cluster of interest to get an estimation of that cluster in the iterative process.

So far we have described the extraction of a latency-variable (shifting) component cluster out of a stationary one, based on the assumption that the latency of the shifting component cluster is known. In real data, however, the latency of shifting components is often unknown. Using RT as a substitute for component latency may not be appropriate because it may not show a fixed time difference to the response, as there may be additional sources to RT variability. Thus, RT is the combined result of several subprocesses, typically including stimulus evaluation (perception), response selection (decision), and motor processes, which all may vary independently in their durations. Indeed, it has been shown that the P300 latency, which is thought to index the timing of stimulus evaluation, shows only a medium-sized correlation with RT (Kutas et al., 1977; for a review, see Verleger, 1997). In some situations a response is not available as a marker, for example, in no-go tasks or in many linguistic studies; nevertheless, the variability of processing stages still exists. As discussed before, previous methods (Takeda et al., 2010; Takeda, Yamanaka, Nozaki, & Yamamoto, 2008) addressed this problem with an optimization strategy, which is impractical because of the large configuration space and many local minima in the presence of noise in the data. To solve such situations, we incorporated an efficient latency-estimation step by examining the cross-correlation between the residue and the estimated component R. We will demonstrate that the latency-estimation step can be inserted into the iteration procedure and achieve a self-improved cycle.

Let us take the simulation model in Figure 1 as an example of how a latency-variable component cluster can be estimated without an explicit marker for a response. Suppose the latency $\tau$ of the component cluster is unknown. We could estimate its latency, for example, by peak detection from the residue in Figure 1B. This estimated latency, say $\tau'$, will deviate systematically from the real value. In reality, there will be an additional error due to noise. However, aligning the residues according to this estimated latency $\tau'$ might still give a reasonable approximation. Using the estimated latency, we can then apply RIDE to obtain the component clusters $S'$ and $R'$, which are not exactly the same as S and R. In the next step, the revised residue EEG minus $S'$ is expected to be closer to R as compared to the initial residues; its latency, estimated from the revised residue by peak detection or by cross-correlation of the new residue and $R'$, is expected to be closer to $\tau$. By carrying out this nested iteration procedure for a sufficient number of steps, we can recover simultaneously the component clusters S and R and the latency $\tau_i$. An obvious question is, now, how much error in the initial estimate of the latency can this method tolerate? We performed a systematic test using our simulation model, assuming that the latency $\tau_i$ has a Gaussian distribution with variance $\sigma_{\tau_i}$. The estimated latency was assumed to deviate from the real value with a random jitter, this deviation being again normally distributed, with variance $\sigma_j$. We studied the performance of RIDE in the parameter space of $(\sigma_{\tau_i}, \sigma_j)$ by measuring the relative error between the real latency and the final latency obtained by RIDE. The results in Figure 1E show that the method is very robust. In a broad region, the component and latency can be recovered precisely if we disregard noise. Of course, it takes fewer steps to converge if the jitter is smaller for a given $\sigma_j$. Only in the region where the distribution of R is strongly localized with small $\sigma_{\tau_i}$ and where jitter $\sigma_j$ in the initial estimation is large will RIDE fail to properly restore the components and their latencies.

Remarkably, the results in Figure 1E suggest that we might simply use randomly selected times as initial latency estimates for R when it is distributed broadly. In Figure 1F, we show the separation performance of RIDE across different levels of noise and compare it with the method by Takeda et al. (2010) and Takeda, Yamanaka, Nozaki, and Yamamoto (2008). The noise was constructed by phase shuffling the prestimulus EEG in the face-priming experiment, which we will analyze in more detail next. Figure 1F shows mean values and error bar (+/-SD) of 50 realizations with reshuffled phase of the noise in each realization. The horizontal axis is the noise level, which we defined as the ratio of the standard deviation of noise over that of the signal R. The vertical axis is the correlation between the estimated latency of R with the given latency. When using Takeda’s method, we performed 50 global random searches with different initial
latencies and then used the best result to conduct the local search (Takeda et al., 2010). In noisy environments, RIDE performed much better for estimating the latencies of R than Takeda's random search method. This was mostly due to the fact that there were too many suboptimal solutions of latency sets, in which the search process was trapped. Moreover, Takeda's method required millions of estimations of latency sets in total, in sharp contrast to RIDE, which only took a few estimations until latencies converged.

Although RIDE and Takeda et al. (2010) and Takeda, Yamana, Nozaki, and Yamamoto (2008) both treat the components' separation and latency estimation simultaneously, the central techniques are markedly different. (a) In component separation, Takeda et al. used Fourier transform to solve the convolution equations and obtain S and R. This method involves the risk of amplifying slow noise and distorting the resulting components. RIDE uses ERP residues and iteration to separate S and R and does not amplify any artifacts. (b) In latency estimation, Takeda et al. used random search, which—as we have shown—performs worse than RIDE (Figure 1F) mostly because of the local minimum problem. RIDE avoids the problem of global optimization of Takeda et al. by employing the residue signal, which can be used to reliably detect the latency in single trials, and converge to the correct components and latencies by iteration. (c) When the latency variability is too small (small \( \sigma_L \) in Figure 1E), RIDE does not work well when the latency is unknown. We also implemented the random search method (Takeda et al., 2010; Takeda, Yamana, Nozaki, & Yamamoto, 2008) and found that this method is also unable to restore the correct components and latencies in this regime. (d) RIDE’s computational effort increases linearly with the number of trials \( N \). In Takeda’s methods, the search of the latency is restricted to \( M \) discrete times (e.g., every 1 ms from 0 to 1000 ms, \( M = 1000 \)). The configuration space increases rapidly with \( N \) as \( M^N \), and in general, the number of local minima of the objective function also increases greatly with \( N \). We compared RIDE and the random search method with the following parameters: \( N = 100, M = 1000, \sigma_L = 100, \) and noise level 1. In Takeda’s method we used 50 global random searches with different initial latencies, each search with five sweeps of all trials. This method used 325,000 latency estimations for the entire process of separation. In sharp contrast, RIDE required only 15 latency estimations to achieve convergence. The analysis above shows that RIDE is very promising for separating single-trial EEG into component clusters with stationary and variable latencies and for measuring these latencies, even when the timing of a response is not available, for example, in no-go tasks. If RIDE works, we might obtain a better understanding of the timing of cognitive (sub)processes in mental chronometry. In the following, we demonstrate the application of our method to a face priming experiment.

In the application, we attempted to separate three major component clusters simultaneously for all the channel by using the same latency for the component across all channels: (a) a stimulus-locked component cluster named S, which is supposed to include neural processes like visual perception or structural encoding, which are relatively unrelated to the response speed and show latencies locked to the stimulus onset; (b) a response-locked component cluster named R, which is supposed to include motor-related processes such as motor preparation and execution; and (c) an intermediate component cluster named C, which is mainly attributed to latency-variable processes and supposed to be related to central cognitive processes, such as stimulus evaluation or response selection. The shapes of S, C, and R and the latency of C are unknown and need to be determined simultaneously by RIDE.

If we have full information of the latencies of S, C, and R, we can use RIDE to separate them. Because we only have two sets of markers—stimulus onset (latency of S) and RT (latency of R)—we have to estimate the latency of C. This estimated latency is used to separate S, C, and R. The separated C is then employed to iteratively improve its latency estimation. To determine the latency of component cluster C in single trials, we used a spatiotemporal template of C. Specifically, for a given subject and condition, we used the separated component cluster C for all recorded channels as a template. We extended the calculation of cross-correlation from one dimension to two dimensions, which we implemented as follows: We combined the time series of all channels to a single time series and compared it to the template (which is also a whole series of all channels in the same manner) by calculating the time-lagged correlation coefficient between them. The time lag giving the maximal correlation was taken as the latency estimate for the component cluster C in single-trial ERPs. Note that, in the beginning, the template of C is not available, and the latency estimation is special for the first time, which will be mentioned below.

The idea of using a spatiotemporal template is motivated by the fact that scalp-recorded EEGs at different channels are highly coherent ensembles because of the underlying neural generator activity and its volume conduction to different recording sites. Therefore, the spatiotemporal pattern of each single trial can reflect an overall latency of brain response. This method can improve the signal-to-noise ratio.

It is also important to realize that the latency obtained in this way need not necessarily correspond to the peak activity at the channel where a component is most pronounced. Because this spatiotemporal cross-correlation yields the common latency for the components of all channels, we can obtain the topography of the separated components. This is one very important advantage of this method in ERP studies, which enables us to obtain better understanding of the cognitive processing, such as the hand-related asymmetry of R and the priming effects in S and C in our example (see below). The method of Takeda et al. (2010) and Takeda, Yamana, Nozaki, and Yamamoto (2008) estimated the latencies of different channels independently, which does not allow assessing the topographical evolution of the separated components. In addition, allowing for decoupling of latencies at different electrodes within a component cluster is very problematic for the concept of an ERP component, which is the effect of a specific brain system that should affect all electrodes at the same moment in time or at most with a (small and) constant delay between electrodes.

The separation steps of RIDE can be summarized as follows:

1. Estimating the latency of C by any kind of method such as peak detection or Woody’s method.
2. Using the estimated latency as the latency of C. Using RIDE to separate S, C, and R based on the three time markers: stimulus onset, latency of C, and RT.
4. Repeating Steps 2 and 3 until the component clusters S, C, and R and the latency of C all converge.
Application of RIDE to an ERP Study of Face Priming

Experiment description. The data to which we applied RIDE were taken from Herzmann and Sommer (2010), where full details about the methods and results from conventional averaging can be found. Briefly, 23 participants made familiarity judgments about famous, unfamiliar, and experimentally learned faces. All stimuli were preceded either by a different face or by the same face; the latter case was a repetition priming condition. Participants indicated target familiarity by key presses with their index fingers. The assignment of familiar and unfamiliar stimuli to the left or right hand was counterbalanced. Prime stimuli were to be ignored.

In the present reanalysis, we used only data for primed and unprimed unfamiliar and famous faces. Only 21 of the original data sets were included because of too many trials with artifacts in the data sets of two participants.

Data preprocessing was as follows: Epochs of 1100 ms, starting 100 ms before the onset of the target face, were generated from the continuous record. Trials with ocular (blINK or saccades) and nonocular artifacts (defined as voltage steps exceeding 50 μV/ms or a difference of more than 100 μV in an interval of 200 ms) and incorrect behavioral responses were discarded. ERPs were aligned to a 100-ms baseline before target onset, digitally low-pass filtered at 30 Hz with zero phase shift, and recalculated to average reference. Data sets contained between 40 and 60 trials for familiar faces and between 124 and 178 trials for unfamiliar faces.

Mean reaction times were shorter for famous (583 ms) than unfamiliar faces (663 ms), $F(1,22) = 20.4, p < .0001$, and for primed (580 ms) than unprimed faces (666 ms), $F(1,22) = 180.6, p < .0001$. The priming effect (RT unprimed minus RT primed) was larger for famous ($M = 135$ ms) than for unfamiliar faces ($M = 36$ ms), $F(1,22) = 23.6, p < .0001$.

Results obtained with RIDE. The ERP data for a typical participant to unprimed unfamiliar faces is shown in Figure 2, with trials sorted according to RT. It appears that the peak of the late P300 or LPC complex shifts in time with RT (Figure 2B), whereas earlier peaks, like P1 and N170, are stationary, best seen at occipital electrodes (Figure 2A).

Figure 2, panels D, E, and F, shows the results of the separated component clusters S, C, and R and some of their features at channel Pz from the unprimed-unfamiliar condition for a typical participant. Note that component clusters C and R in Figures 2E and 2F are displayed by setting their latencies to 500 ms. Features of the component clusters are similar across participants and conditions. Component cluster S is similar to the average stimulus-locked ERP (Figure 2B) and retains most features such as the earlier peaks and the basic shape of the LPC. The other two component clusters increasingly lose the features of stimulus-locked ERPs; component cluster R especially displays a distinct time course completely different from stimulus-locked ERP components and has a relatively small amplitude. This result is in line with the idea that the conventionally averaged ERP locked to one event will seriously distort the components locked to other events by smearing and flattening them. Figure 2G–I displays the S, C, and R component clusters, respectively, for all single-trial ERPs sorted by RT, when the other two component clusters are removed at their detected latencies. According to our assumption, component cluster S should have a fixed latency relative to the stimulus onset, whereas C has varying latencies relative to the stimulus, and R should be synchronized with RT. This result is not so obvious in the color plot of each component cluster in single-trial ERPs at Pz because of strong background noise, but it is very pronounced in the corresponding spatiotemporal cross-correlations (STCC) between the component template and single trials (Figure 2J–L). The latencies (Figure 2M–O) calculated from the peak of STCC in Figure 2J–L again support the assumption. Indeed S has a fixed latency irrespective of strong variability in RT. C has a variable latency which is correlated with RT ($r = .60$). R is supposed to be very closely locked to RT. Indeed, there is a maximum of STCC very close to RT in almost all trials. However, as seen from Figure 2F, the amplitude of R is small and its peak is narrow (about 3 Hz); therefore R in some trials (about 20%) may be strongly affected by noise. In these trials noise contamination might produce an ambiguous peak of STCC away from RT, but of larger value than the peak close to RT. Therefore, by following the maximal value of STCC as shown in Figure 2O, a few scattered points lead to a low correlation with RT ($r = .39$). If we take out these 20% ambiguous trials, the correlation coefficient significantly increases to $r = .98$.

The results for a single subject, shown in Figure 2, are quite similar to the other subjects. To demonstrate the robustness of RIDE across participants, we show the time course of the separated components S, C, and R at Pz for all subjects in the unprimed-unfamiliar condition (Figure 3A–C). We then calculated the correlation between the time course of each subject and the grand mean over all subjects, as given in the figure. The results demonstrate the consistency of the template across subjects. The right panel (Figure 3D–F) shows the cross-correlation between the spatial-temporal template of S, C, and R with the single trials (with the other two component removed) for all trials from all subjects in the unprimed-unfamiliar condition. The trials are sorted by RT.

These results show clearly that not only are the components obtained by RIDE robust across subjects, they are also consistently present in most single trials. Even for the small component R, which can be easily contaminated by noise, we can see that in most of the trials the maximal correlations occur very close to the response (Figure 3F). Later we will show that condition effects in component clusters S and C are also robust across subjects.

Validation of RIDE. Consistent with previous models (Hansen 1983; Takeda et al., 2010; Takeda, Yamanaka, Nozaki, & Yamamoto, 2008; Takeda, Yamanaka, & Yamamoto, 2008; Zhang, 1998), RIDE is based on the assumption that a single-trial ERP is a linear superposition of stationary components with latency-variable components. The theoretical framework itself cannot provide information to judge whether assumption and model are appropriate, and previous works (Hansen 1983; Takeda et al., 2010; Takeda, Yamanaka, Nozaki, & Yamamoto, 2008; Takeda, Yamanaka, & Yamamoto, 2008; Zhang, 1998) did not provide evidence to validate such a model. However, we can examine the properties of the data to check if the model is consistent with or at least provides a good approximation to the reality. We have therefore carried out several tests: (a) testing the overlap between static and shifting components in the data through suitable grouping; (b) comparing results from real data with an alternative assumption that a single-trial ERP consists of only one component, whose width expands with RT; (c) applying various latency detection methods to real data and simulated data, which were constructed from templates S and C using linear
Figure 2. Separation of components for the unprimed, unfamiliar condition of a typical participant. A: Color plot of single-trial ERPs sorted by RT (black line) at channel PO8. B: Color plot of single-trial ERPs sorted by RT (black line) at channel Pz. C: Time course of traditional averaged ERP at channel Pz. D–F: Time course of separated component clusters S, C, and R. In this presentation, the peaks of C and R are put at $t=0$ ms. Note that the amplitude of R is much smaller than S and C. G–I: The corresponding color plot of one component cluster in single trials with the other two component clusters removed, again sorted by RT. J–L: The corresponding spatiotemporal cross-correlation between template S, C, R, and the single trials shown in G–I, respectively. M–O: The corresponding peak latencies obtained from J–L, plotted with respect to the RT of the each trial. Because R is weak, a noise component with a pattern similar to that in R is contained with a certain probability in the EEG trials. This will cause distractive peaks (about 20%) in the cross-correlation in L.
superposition models; if there is a significant nonlinear interaction between the components, the results from the real data should most likely deviate systematically from the linear model; and (d) examining the functional significance and independence of the separated component clusters.

**Testing the overlap between different components.** In the following subsections, we describe the latency comparison and functional independence of the components. First we present the results for the testing of the overlap between components. For this purpose, we examined the two large component clusters S and C. After obtaining the latency $\tau_i$ for C in Figure 2N, we sorted the trials according to $\tau_i$, divided them into bins according to $\tau_i$, and averaged them to reduce background noise. As can be seen in Figure 4A, for trials with the smallest C latency, there is only one peak around 500 ms where the peaks of C and S overlap. For intermediate latencies, two peaks start to emerge, and, for the largest latencies, the two peaks are clearly separated. However, in all cases, the first peak is fixed around $t \sim 500$ ms. The effects were best seen at electrode PO3 in this participant but were also present at other sites. These results provide very direct evidence of the superposition of a stationary component and a moving one.

**Comparison of real data with alternative models.** The additivity assumption is further reinforced by rejecting an alternative assumption that the LPC is an inseparable ensemble, for example, of a single component broadening in width with increasing RT. We designed such an “expanding” model, putting C, S, and noise into the single trial, of which C has an expanding width correlated with latency. The expanding principle was realized by locking the left end to Time 0 and pulling the peak of the C template to the point of latency (from RIDE of real data), so that the width of C was determined by its latency. Then we used RIDE to separate the expanding model, which produced another S' and C'. We estimated the width of C' in each single trial by subtracting S' and normalizing the amplitude; then the width of C is proportional to the area $A$ of each single trial with respect to the baseline, according to the formula $A \sim height \times width$. The result shows that the width of C derived from the expanding model did increase with latency whereas it was very stable in the real data (Figure 4B). This analysis provides further validation of the linear superposition assumption for the real data.

**Comparison with conventional latency detection methods.** The separation results indicate that the P300 (or LPC) consists of a stationary component and a shifting component, which questions methods that—implicitly or explicitly—consider P300 as a single component, such as Woody’s method, peak detection. Applying these methods to latency detection may produce systematic errors. To demonstrate this, we compared the results obtained from these methods with those of RIDE.

We sorted all the trials from the data of the exemplary participant on channel Pz according to the latency of C. For comparison, we also constructed a set of simulated ERP data using the component cluster S and C and the corresponding latency $\tau_i$, based on the linear superposition model, $EE_{\tau_i}(t) = S(t) + C(t + \tau_i)$ without considering noise. All methods were applied to simulated and real data and compared with each other. Note that in this simulated data, component clusters S and

---

**Figure 3.** A–C: The time course of the separated component cluster S, C, and R at Pz for all subjects in the unprimed-unfamiliar condition (amplitudes were normalized to the average level). D–F: The spatiotemporal cross-correlation between S, C, R, and single trials (other components removed) for all trials across all subjects (sorted by RT; dashed line). The peaks of C and R were adjusted to 0 ms so the peaks in E and F correspond to the peak latency of C and R with respect to the stimulus onset.
C are present in all trials. However, if there is strong nonlinearity in real data, violating the additive model, single-trial ERPs can differ systematically from the simulated data, which would most likely be manifested in a systematic deviation of their latencies from the simulated data.

The results are shown in Figure 5. We used Woody’s method (Figure 5A) and peak detection by selecting the most positive point between 200 and 1000 ms (Figure 5B) to detect the latency of single trials. Note that in order to show clearly the comparison with simulated data, we had to smooth the trials (grouping and averaging 10 trials as a new trial and moving 1 trial forward). We applied peak detection to the residues (Figure 5C) and detected the latency by cross-correlation between the template C (not spatiotemporal, but only at Pz) obtained by RIDE and single-trial data with component cluster S removed (Figure 5D).

Figure 5 shows that RIDE obtains results similar to those of each of the other methods from both the original (dots) and simulated data (blue bold line). This cannot be true if the model is a poor descriptor of the real EEG and supports the linear superposition model as a good description of single-trial ERP data, especially, that they can be regarded as consisting of temporally independent component clusters S and C.

The methods illustrated in Figure 5A–C produced systematic errors when compared to the real latency \( t \) of the shifting com-

Figure 4. A: Evidence of the superposition of stationary and moving components in the EEG. ERP trials at PO3 are sorted into three bins according to the latency of C in Figure 2N, grouped, and averaged. B: The estimated width of C for all trials (sorted by latency). The ERP data were smoothed by grouping 20 trials.

Figure 5. Vertical axis: latency of C obtained from RIDE. Horizontal axis: latency detected by different methods from real ERP data (dots) and simulated data (blue line) on channel Pz. A: Woody’s method. B: Peak detection from single-trial ERPs, smoothed over 10 trials. C: Peak detection from ERP residues. D: Cross-correlation between template C and single trials with other component clusters subtracted. The result was smoothed by grouping 10 trials.
ponent cluster C. Woody’s method yielded a much narrower distribution, which was biased to the peak of the component cluster S at around 500 ms, because S contributed significantly to the cross-correlation between single-trial ERP and the template, which mixes stationary S with shifting C. The peak-detection method identified long latencies well when the peak of C was much later than that of S (Peak 2 in Figure 4), but systematically produced overly large values when τ was small and the peak in the single trial is biased to that of S. The latency detected from the residues differed from τ because the residues represent a distorted version of the variable-latency component cluster C (cf. Figure 1B and Equation 3). The latency will be detected properly in the single trial only after decomposing the latency-variable component cluster C from the static component cluster S (Figure 5D).

In sum, applying several alternative methods other than RIDE to real and simulated data (S+C, linear model) led to the same results. This is support for the linear superposition assumption. We can thus assume that the latency derived by RIDE is close to the real one and that other methods produce systematic errors.

**Examining the functional significance and independence of the separated component clusters.** In this section, we show that the component clusters separated by RIDE are also consistent with cognitive understanding and expectations. This will provide further justification of the model.

The distinct time courses and latency distributions suggest that the component clusters are indeed different and independent. The upper panel of Figure 6 shows the time course of the different component clusters at representative channels, averaged over all conditions, right-hand responses, and all participants. Each component cluster clearly has its own scalp topography, indicating that they are generated by distinct neural sources. In particular, S at around 100 ms shows a center at the occipital electrodes O1 and O2 (related to visual processes), C has a focus at Pz (related to stimulus evaluation), and R peaks at Cz and C3 around the RT (related to right-hand motor responses). This shows that ERP components are reflected in the component clusters; for example, visual processing as reflected in the positivity at electrodes O1 and O2 appears only in component cluster S but not in the two others.

Because the component cluster R is locked to RT, it is most likely associated with motor-related processes and should exhibit hand-related asymmetry over the scalp, whereas this should not be the case for component clusters S and C. Figure 6, middle panel, provides support for that claim by showing the evolution of the topography for all component clusters over time. The asymmetry of component cluster R begins around 150 ms before the response, with more negativity appearing on the contralateral side. This result is consistent with the contralateral negativity during the preparation of a lateralized movement (Coles, 1989). The asymmetry pattern continues while the scalp topography evolves. The strongest asymmetry appears about 20 ms after the button press, probably reflecting the motor potential (Tarkka & Hallett, 1991). In another data set to be published elsewhere, we confirmed that such hand-related, asymmetric activity is not found in no-go trials, where no motor response is required.

The independence of the three separate component clusters can also be supported by distinct time-frequency representations using wavelet analysis of the single-trial data in Figure 6, bottom panel. The time-frequency pattern of the original data on channel Pz shows a frequency structure dominated by low frequencies. The results show significantly different patterns when the wavelet transform was applied to single trials that contain only one component cluster. For these single trials, the other two component clusters were subtracted and the time-frequency coefficient was aligned to the corresponding latency and averaged. Now, each component cluster has a different dominant frequency range that is not revealed at all from the original data. S centers at 0.5 Hz, C around 1.0 Hz, and R at the range of 1 to 6.0 Hz.

**Application to face priming.** Separation of components from the mixture in the averaged ERP is especially meaningful when comparing brain activity under different cognitive conditions. After separation, ERP effects can be attributed more specifically to cognitive factors and can be related either to amplitude changes of particular component clusters, their latency variation, or both. More importantly, the differences caused by amplitude and latency effects could have different scalp topographies. Thus, ambiguity and possibly misleading interpretation of the conventional analysis of ERPs might be resolved. In the following, we demonstrate that the separation of component clusters can provide a better understanding of priming effects in familiar face priming when applied to a full data set from a group experiment.

The conventional ERP analysis by Herzmann and Sommer (2010), performed on data from originally 23 participants, showed four different priming effects between 250 and 500 ms. These priming effects were also present in the current subsample (Figures 7A–C and 8A–D) of 21 participants. The first two effects (Figure 8A, B) were considered as variants of the early repetition effect (ERE/N250r), interpreted as electrophysiological indicator of the access to structural representations of familiar faces (Schweinberger, 2011). The last two priming effects were seen as variants of the late repetition effect (LRE; Schweinberger, Pütze & Sommer, 1995), interpreted as an indicator of the access to biographical knowledge or semantic information about familiar people. However, there are several peculiarities in the data of Herzmann and Sommer but also in many other priming studies. First, the scalp distribution of the ERE/N250r was somewhat different from other studies of face priming, which often show a more frontally distributed positivity and more temporal negativity (e.g., Engst, Martin-Loeches, & Sommer, 2006; Herzmann et al., 2004; Schweinberger et al., 1995). Thus it is possible that the manifold of different repetition effects found within a short period of time results from the overlap of just two components, which develop in parallel but with different gradients, yielding different mixtures of these components at different time segments (Schweinberger et al., 1995). Second, as in many other studies of priming, one wonders whether the later part of the repetition effect may be a result of latency shifts between the primed and unprimed conditions and different latency variability. Thus, the parietal positivity in the primed condition is earlier and broader than in the unprimed condition (Figure 7c). To resolve these issues, we applied RIDE to these data. We expected that RIDE would be able to separate early and late repetition effects and clarify the contributions of latency shifts and temporal smearing to the observed priming effects.

First, we found that the priming-related time shift of the peak in averaged ERPs (best seen at Pz) is very similar to the average latency shift of the component cluster C relative to the stimulus as shown by the red and blue bars in Figure 7C. Indeed, when the pure component cluster S is considered (Figure 7D–F), the la-
Figure 6. Upper panel: The time course of the three separated component clusters S, C, and R on seven representative channels, grand averaged over all trials in all four experimental conditions for right-hand responses in all participants. The three topographic patterns represent the scalp distribution of different component clusters at the time indicated by the arrows. Middle panel: The evolution of scalp topographies over time of all three component clusters for left-hand (L) and right-hand (R) responses, based on grand averaged component clusters in all four experimental conditions. Lower panel: The time–frequency pattern of (A) the original ERPs on channel Pz, (B) single trials containing S only, (C) single trials containing C only (aligned to C latency and averaged), and (D) single trials containing R only (aligned to RT and averaged). Here we used the complex morlet wavelet as the base. Note that the latency-variable component clusters C and R and their time–frequency patterns are aligned at their latencies at $t = 500$ ms. This means that for R, $t = 500$ ms corresponds to RT.
tency shift at the parietal electrode is diminished. This is shown by the red and blue bars in Figure 7F, which depict the average latency of S with respect to the stimulus. Panels G to I show the component cluster C for the primed and unprimed conditions, locked to the stimulus.

Interestingly, component clusters S and C both show priming effects. These priming effects can be compared best to those in the conventional ERPs when their topographies are plotted for the same time segments. The top row in Figure 8 shows the priming effects for familiar faces in the conventional ERPs as reported by Herzmann and Sommer (2010).

Statistical analyses of these time segments, slightly different from the original analysis, confirm the previously reported findings (Herzmann & Sommer, 2010) for the present subset of participants. The overall, Huynh-Feldt (1976)-corrected analysis of variance (ANOVA) included the factors time segment (ERE1, ERE2, LRE1, LRE2), electrode (65 channels), and priming (primed, unprimed). For mean amplitudes, this analysis yielded a significant time Segment × Electrode × Priming interaction, $F(192,3840) = 16.4, p < .0001, \epsilon = .043$. Please note that because the average reference sets the mean activity across all electrodes to 0, condition effects are only meaningful in interaction with the electrode factor. A Bonferroni-corrected posttest showed that priming effects were significant in all time segments, $F$s(64, 1280) = 15.8, 33.6, 30.9, and 9.6, $ps < .0001, \epsilon$s = .073, .078, .078, and .075, for the ERE1, ERE2, LRE1, and LRE2, respectively. Paired comparisons between time segments yielded significant differences between ERE1 and ERE2, $F_s(64,1280) = 21.2, p < .0001, \epsilon = .077$, ERE2 and LRE1, $F(64,1280) = 5.1, p < .01, \epsilon = .050$, and LRE1 and LRE2, $F(64,1280) = 22.0, p < .0001, \epsilon = .078$. Differences between topographies of priming effects in adjacent time segments were confirmed by scaling the difference waveforms (primed minus unprimed) for each participant to the same overall amplitude, with the average distance of the mean, derived from the individual mean ERP, as divisor (McCarthy & Wood, 1985). ANOVAs comparing scaled amplitudes yielded significant differences between ERE1 and ERE2 topographies, $F(64,1280) = 6.9, p < .0001, \epsilon = .075$, ERE2 and LRE1, $F(64,1280) = 5.1, p < .001, \epsilon = .063$, and LRE1 and LRE2, $F(64,1280) = 12.3, p < .0001, \epsilon = .097$. These statistical analyses show that all four priming effects are separable with regard to mean amplitudes and topographies.

**Figure 7.** Priming effects in familiar face recognition. A–C: Stimulus-locked grand-averaged ERPs in response to famous faces for the primed and unprimed conditions. The red and blue bars show the average latency of component cluster C separated by RIDE for the primed and unprimed conditions, respectively. D–F: Grand-averaged component cluster S for primed and unprimed conditions. The red and blue bars denote the average latency of component cluster S for the primed and unprimed conditions, respectively. G–I: Grand averaged component cluster C for primed and unprimed conditions, locked to the stimulus.
The priming effects in component cluster S (Figure 8, middle row) start with a topography that resembles more closely the topographies that have been published as ERE for familiar faces (e.g., Herzmann, Schweinberger, Sommer, & Jentzsch, 2004) than does the ERE in the present study. Furthermore, this ERE-like topography is maintained over at least 200 ms. At no time does it resemble the LRE, which would be characterized by a parietal positivity. The overall ANOVA (following the same statistical testing as for the conventional ERP analyses; see previous paragraph) showed a significant Time Segment × Electrode × Priming interaction, $F(192,3840) = 8.6, p < .0001, \varepsilon = .030$. A posttest yielded significant priming effects for the first three time segments, $F(64,1280) = 4.8, 5.7, \text{and } 4.3, p_s < .01, \varepsilon_s = .039, .041, \text{and } .049$, for the ERE1, ERE2, and LRE1, respectively, but not for the last segment, $p = .26$. Paired comparisons between time segments showed a significant difference between ERE1 and ERE2, $F(64,1280) = 10.4, p < .0001, \varepsilon = .071$. ERE2 and LRE1 did not differ significantly from one another, $p = .104$. ANOVAs comparing scaled amplitudes also yielded a significant difference between ERE1 and ERE2 topographies, $F(64,1280) = 4.8, p < .01, \varepsilon = .067$. These analyses show that component cluster S captures two priming effects, one from 250 to 310 ms and a second from 310 to 430 ms, which differ from each other in mean amplitudes and topographies.

Priming effects for the latency-uncorrected, that is, stimulus-synchronized C (Figure 8I–L) consist of a parieto-occipital positivity accompanied by a frontal negativity. In contrast to the priming effect in S, which changes over time, the priming effect in C seems to remain relatively stable. The overall ANOVA confirmed this impression by yielding a significant main effect of priming, $F(64,1280) = 3.9, p < .05, \varepsilon = .034$, but no significant Time Segment × Electrode × Priming interaction, $p = .193$. A posttest showed significant priming effects in all time segments, $F(64,1280) = 3.4, 4.2, 4.6, \text{and } 3.1, p_s < .05, \varepsilon_s = .030, .030, .035, \text{and } .044$, for the ERE1, ERE2, LRE1, and LRE2, respectively. This result shows that component cluster C captures a single priming effect, which is characterized by a parieto-occipital positivity.

We also conducted further ANOVAs to test the difference between component clusters S and C in topographies. For scaled amplitudes in each time segment separately, we calculated an ANOVA including the factors electrode (65 channels), priming (primed, unprimed), and component cluster (S, C). These analyses confirmed significantly different topographies for component clusters S and C in the ERE1 time segment, $F(64,1280) = 4.6, p < .001, \varepsilon = .068$, the ERE2 time segment, $F(64,1280) = 3.7, p < .01, \varepsilon = .066$, and the LRE1 time segment, $F(64,1280) = 2.4, p < .05, \varepsilon = .071$. The difference between priming effects in the LRE2 time segment was not analyzed, because the priming effect in component cluster S for that time window was not significant. These analyses showed that component clusters S and C are indeed different neural correlates of priming with distinct underlying neural sources.

From these observations and statistical analyses, we can draw a first conclusion: RIDE is able to distinguish between priming effects more clearly than conventional ERPs. It is able to separate the LRE, which appears to be linked to the time-variable component cluster C, from other priming effects. When synchronizing C to the stimulus, we obtain a priming effect with parieto-occipital positivity, closely resembling the topography of the LRE. The latency-invariant component cluster S seems to reflect two priming effects, which are very likely related to the ERE. Recent research has suggested two contributing sources of the ERE: perceptual priming and priming of memory representations (Böhm, Klostermann, & Paller, 2006; Dörr, Herzmann, & Sommer, 2011). RIDE appears to be able to extract these different priming sources by showing a possible perceptual priming effect from 250 to 310 ms and a possible memory-related...
priming effect from 310 to 430 ms within component cluster S. It is
impossible to imagine that the sum of the priming effects in S and
C in each time segment yields the priming effect in the conven-
tional ERP. Indeed, the sum of S and C shows priming effects not
topographically different from the conventional ERP analysis,
$F(64,1280)<0.28$, $p_5=.99$, .96, .98, and .99, $t_5=.103$, 115,
.117, and .113, for the time segments of the ERE1, ERE2, LRE1,
and LRE2, respectively, using the same ANOVA as for the
comparison of S and C (see previous paragraph).

Although RIDE seems to be able to separate these component
clusters, the question remains of how exactly time shifts and
differences in latency variability contribute to the emergence of a
priming effect in C amplitude. This can be investigated by look-
ing at C amplitudes when latency shifts and variability are com-
penated by synchronization at the maximal cross-correlation
with the template. The results are shown in Figure 9. The two top
rows show the primed and the unprimed conditions. In the un-
primed condition, a parieto-occipital negativity is visible first,
lasting for about 200 ms, followed by a parietal positivity. The
primed condition, in contrast, mainly shows the parietal posi-
tivity to various degrees. Thus, priming reduced the parietal neg-
ativity. When the priming effect was calculated, as common, by
subtracting the unprimed from the primed condition, a parieto-
occipital positivity emerged. This parieto-occipital positivity is
quite similar to the priming effect in the latency-uncompensated
(stimulus-locked) C component cluster.

We may therefore conclude that the LRE is not an artifact of
latency shifts and different degrees of smearing because it is
present also in latency-compensated C wave shapes. If anything,
it seems to be in the latency-compensated than in the stim-
ulus-locked signals; therefore the LRE is not present because of
latency variability and smearing but despite these detrimental
effects.

Another interesting property of component cluster C is the
parietal negativity in the unprimed condition. It is an old sug-
gestion that the increased positivity in primed faces, but also in
words or other stimuli, is due to the reduction of an N400 com-
ponent that occurred in the unprimed condition (Barrett & Rugg,
1989). However, this argument was difficult to prove because the
negativity, which is supposed to be reduced by priming and to
lead to an increase in positivity, was hard to show. In Figure 9B,
we can now clearly see this negativity, which we suggest to reflect
the N400 component. This N400 component is contained in C
and is indeed abolished by priming.

In sum, RIDE has allowed us to separate different priming
effects and to indicate their mode of superposition. RIDE has
also shown that the LRE is not an artifact of different degrees of
latency shifts and variability. Finally, it has helped to visualize a
hitherto often hypothetical component.

Discussion

The Achievements of RIDE

In this article, we introduced a novel analysis method for ERP
data, RIDE, with the aim of revealing a clearer picture of ERP
composition and clarifying some ambiguities inherent in con-
ventional methods. RIDE is in line with the linear superposition
model (Hansen, 1983; Takeda et al., 2010; Takeda, Yamanaka,
Nozaki, & Yamamoto, 2008; Takeda, Yamanaka, & Yamamoto,
2008; Zhang, 1998). It decomposes single-trial ERPs into static-
latenacy and variable-latency components associated with various
stages of cognitive processing and can reliably detect the latency
of these components in single trials by applying a spatiotemporal
template of the components. By using the residue of EEG in an
iterative manner, RIDE solves the difficult problem of estimating
the latencies in single trials together with the components of
ERPs in an efficient way, avoiding the uncertainty in the opti-
mization scheme and artifacts produced by deconvolution-based
methods using Fourier transform (Takeda et al., 2010; Takeda,
Yamanaka, Nozaki, & Yamamoto, 2008; Takeda, Yamanaka, &
Yamamoto, 2008). RIDE uses the spatiotemporal cross-correla-
tion to estimate the component topography in single trials at a
common latency for all electrodes, which is another important
advantage for application to ERP studies. RIDE can thus reveal
distinct stages of neurocognitive processing with separate time
courses and topographic patterns. This separation method offers
a new way to more precisely estimate timing and identify cog-
nitive stages, which is not possible for traditionally averaged
ERPs that intermix multiple components. By separating static-
latenacy and variable-latency component clusters, we can study
the impact of manipulations on various cognitive processing
stages. We have demonstrated that the methods can separate the
effects of amplitude and latency variations in ERPs for primed
and unprimed conditions, allowing much better understanding
of the cognitive processes. In cognitive experiments, different
conditions will typically lead to changes both in brain activity

Figure 9. Repetition priming effect in component complex C, aligned to the maximal cross correlation with the template. A: primed condition. B:
Unprimed condition. C: Difference waves (primed minus unprimed).
and response speed. This method has, therefore, great potential for numerous applications.

Any method of component separation requires some more or less arbitrary decisions at least about the number of components into which the variance in the data is separated. For example, ICA separates a great number of components (up to the number of electrodes), which have specifically localized and independent sources. However, it is difficult to determine, which, if any, components correspond to each other across participants. RIDE does not separate ERPs into individual components corresponding to independent sources inside the brain, but into component clusters that could contain multiple components and sources, which reliably occur across trials and have relatively similar roles; for example, they are temporally synchronous, locked to the same event or functionally distinguished mental stages. In this work, we considered three component clusters corresponding to stimulus-locked, response-locked, and central components. We found that the results are consistent with and, even more importantly, can be applied to ERP analysis (priming effects) and provide novel solutions for hitherto unresolved problems. In principle, and as discussed in the next section, RIDE can be extended to even more component clusters than three in two-marker data. Similarly, the ERP in no-go data with only one marker can be separated into two or more component clusters.

Clarification and Some Limitations of RIDE

There are some unavoidable limitations and boundaries for the application of RIDE. They are as follows:

1. The assumption. RIDE assumes a linear superposition of different component clusters. This seems to be a good approximation to the data analyzed here. This assumption, however, may bias the results when strong nonlinear interactions between the component clusters are present.

2. Noise and artifacts. Too much noise or too many artifacts will certainly affect the correctness of the result, especially during the latency detection step. A strict artifact removal in preprocessing of the data is necessary. A sufficient trial number is important for obtaining stable and reliable results in the separated ERP templates and the latencies in single trials.

3. Latency variability. Although latency variability is present in any kind of cognitive experiment, the distributions may vary strongly. Too narrow distributions are not good for the separation because the result will be more sensitive to the initial condition or always converge to different results due to different initial conditions.

4. Number of component clusters. Here we separated ERPs with two time markers into three component clusters, which are marker-locked or -unlocked. In principle, RIDE could further separate the marker-unlocked cluster, but at present we have not yet developed criteria for determining the maximum number of possible component clusters. These criteria might be based on the similarity and reliability of the separated component clusters across single trials; their development is a topic for further exploration.

It is important to point out again that each component cluster obtained by RIDE is likely to contain several ERP components that move together in time but may be generated by different sources and reflect different functions.

Further separation of these time-locked components within each component cluster could be achieved, for example, with ICA.

Possible Applications and New Directions

RIDE is a method that is useful whenever there are different degrees of latency variability in ERP components. It can help to separate components that would otherwise overlap with each other. For example, RIDE may be a useful tool to disentangle separate stimulus- and response-related and central contributions to the P3 component or late positive complex.

Because RIDE yields information about the timing of component clusters, it may shed light on basic questions of mental chronometry like discrete versus continuous or serial versus parallel processing.

RIDE may be especially useful when differences might be caused by different degrees of latency variability or time shifts, as we have demonstrated above for the condition differences in priming effects. But it could also be found in differences between populations. For example, patient populations may differ in the amplitude of a given component not because the components are indeed smaller but because they are merely more variable in their timing.

Interestingly, RIDE is not confined to designs that require overt responses and yield reaction times. In fact, RIDE can be applied even in no-go tasks as we will show elsewhere. We believe our method can provide a better understanding of mental chronometry with ERPs compared to previous methods. This offers unique perspectives on many data sets recorded in situations without (immediate) overt responses, as, for example, in psycholinguistics when the response is not speeded (i.e., there is an emphasis on accuracy), not required in all trials, or complex (like ratings, verbal reports).

References


RIDE: A new method to separate ERP components


(Received May 8, 2011; Accepted June 14, 2011)