British Journal of Mathematical and Statistical Psychology (2018), 71, 146–166

The British Psychological Society

www.wileyonlinelibrary.com

Mathematical transcription of the 'time-based resource sharing' theory of working memory

Nicolas Gauvrit[1]* and Fabien Mathy[2]

[1]Human and Artificial Cognition Lab, Practical School of High Studies, Paris, France
[2]University of Nice Sophia Antipolis, France

The time-based resource sharing (TBRS) model is a prominent model of working memory that is both predictive and simple. TBRS is a mainstream decay-based model and the most susceptible to competition with interference-based models. A connectionist implementation of TBRS, TBRS*, has recently been developed. However, TBRS* is an enriched version of TBRS, making it difficult to test general characteristics resulting from TBRS assumptions. Here, we describe a novel model, TBRS2, built to be more transparent and simple than TBRS*. TBRS2 is minimalist and allows only a few parameters. It is a straightforward mathematical transcription of TBRS that focuses exclusively on the activation level of memory items as a function of time. Its simplicity makes it possible to derive several theorems from the original TBRS and allows several variants of the refresh process to be tested without relying on particular architectures.

1. Introduction

Working memory is often described as a unique cognitive resource serving both short-term maintenance and processing (Baddeley, 2003). Thus, it is central in problem-solving (Swanson & Beebe-Frankenberger, 2004). Working memory is known to be mediated by the prefrontal cortex (Braver & Bongiolatti, 2002; Kane & Engle, 2002; Prabhakaran, Narayanan, Zhao, & Gabrieli, 2000) and is clearly linked to intelligence (Conway, Cowan, Bunting, Therriault, & Minkoff, 2002; Engle, Tuholski, Laughlin, & Conway, 1999), particularly when complex span tasks are used (Unsworth, Redick, Heitz, Broadway, & Engle, 2009).

The *complex span task* paradigm is probably the most widespread dual procedure for measuring working memory (Conway, Kane, Bunting, Hambrick, Wilhelm, & Engle, 2005). In complex span tasks, participants are presented with items to be memorized, and instructed to recall them at the end of the trial in the correct order. In contrast to simple span tasks, the presentation of the memory items is interspersed with a concurrent task that presents distractor items. Participants thus alternate between encoding memoranda and processing distractors.

A variety of concurrent tasks have been used, including reading sentences (van den Noort, Bosch, Haverkort, & Hugdahl, 2008), reading digits (Barrouillet, Bernardin, & Camos, 2004), verifying arithmetic statements (Redick *et al.*, 2015; Turner & Engle, 1989), uttering predetermined syllables (Lewandowsky, Geiger, Morrell, & Oberauer,

*Corresponding should be addressed to Nicolas Gauvrit, Laboratoire CHArt (EA 4004), Ecole Pratique des Hautes Etudes & Université Paris-Saint-Denis, 4-14 rue Ferrus, 75014 Paris, France (email: ngauvrit@me.com).

DOI:10.1111/bmsp.12112

2010), and diverse elementary tasks involving executive functions such as memory retrieval, response selection, and updating (Barrouillet, Portrat, & Camos, 2011). A critical feature for determining the validity of complex span tasks is the degree of control the experimenter has over the time devoted to processing the distractors or encoding the memory items (Oberauer & Lewandowsky, 2011). In this respect, a computerized version of a complex span task (Barrouillet & Camos, 2007) offers fine control over a participant's processing timeline.

In a computerized version of the complex span task, participants are, for instance, required to memorize a series of items appearing on a screen sequentially. Each item is displayed once for a fixed amount of time (often fixed to 1 s). Between these memory items, participants must perform a second task. For instance, for the second task, a participant could see a '2' for a short period, then '+3', then '−1', and be required to update the digit 2 into 5 and then 4. At each step, the participant is instructed to verbalize the current result of the series of arithmetic operations. At the end of the trial, the participant must recall the items to be memorized in order at their own pace. A trial for a list of two letters would look like this: L, 2, +3, −1, H, 3, +1, −2, Recall. A correct response would be 'LH' in that case.

Barrouillet *et al.* (2004) developed the *time-based resource sharing* (TBRS) model to specifically account for the performance of participants in such experiments. The key ideas First, unless attention is focused on the memorandum to refresh the memory items, their memory traces fade away. Second, because of a bottleneck effect, attention can be devoted only to either refreshing the memorandum item by item or processing the distractors. Thus, participants perform rapid switches between refreshing items and performing the concurrent task. Third, the probability that an item is correctly recalled is a function of the item *cognitive load*, defined as the proportion of time that cannot be devoted to refreshment of the item. Fourth, temporal factors (instead of the competition between items) are preponderant, especially in tasks designed to limit interference (Barrouillet & Camos, 2012; p. 414), even though interference is acknowledged and studied within the model (Camos, Mora, & Barrouillet, 2013; Portrat, Guida, Phénix, & Lemaire, 2016).

Time-based resource sharing has received much qualitative empirical support (Barrouillet & Camos, 2012; Barrouillet, Gavens, Vergauwe, Gaillard, & Camos, 2009; Portrat, Barrouillet, & Camos, 2008; Vergauwe, Barrouillet, & Camos, 2009). However, although it is both simple and rooted in a set of clearly stated hypotheses, the model remains underspecified. First, it does not indicate how long the refreshment period (between two switches) lasts for an item and whether this duration is fixed or determined by specific factors (but see below for more details about this issue). It also does not indicate what happens when the refreshment has been interrupted by a distractor, that is, whether people start refreshing the first item anew or continue from the last refreshed item, and so forth. Second, although TBRS is described in detail in several publications (Barrouillet & Camos, 2007; Barrouillet *et al.*, 2004), it has remained a theory based mostly on a verbal description (until recently; see below). This is unfortunate because it reduces opportunities to test the model and use statistical criteria of fit (Pitt & Myung, 2002). Moreover, because building a mathematical description of TBRS is easy and straightforward, verbal descriptions alone should be avoided (Farrell & Lewandowsky, 2010; Norris, 2005; Pothos & Wills, 2011).

Oberauer and Lewandowsky (2011) have developed the only available computational implementation of TBRS, called *TBRS**, a two-layer connectionist network. One objective of the authors was to bridge existing computational models of working memory with prominent features of TBRS. Although the model was found to fit experimental data, the authors remain sceptical about TBRS in their conclusion.

TBRS* is an important step towards a precise quantitative validation of TBRS; however, it may be a model that is too enriched compared to the original TBRS. Some caveats should therefore be kept in mind. First, TBRS* merges features from TBRS and other models. Hence, if it fails at predicting empirical data, it would be unclear whether this failure should be taken as evidence against TBRS or against other features related to the specific implementation. Second, because TBRS* is an enriched model, some decisions were made (e.g., serial position coding) that would be unnecessary for defining a more basic implementation of TBRS. Last, connectionist networks can be seen as 'black boxes' and may lack transparency (Shahin, Jaksa, & Maier, 2009), making it difficult to formally demonstrate results that nonetheless follow from the TBRS framework.

Here, we present a new mathematical implementation of TBRS (henceforth TBRS2) designed to remain as close as possible to the TBRS assumptions. The approach developed here is radically different from the one underlying TBRS*.[1] On the one hand, our implementation is not as rich as TBRS* and cannot account for as many features, as we do not address any question not already addressed in the TBRS verbal description. For instance, we do not aim to code the order of items in any way, but stick to predicting the correct recall of each item. On the other hand, TBRS2 has two interesting features. First, we must only determine the decay function of memory traces and schedule of refreshment, that is, two decisions; whereas TBRS* must make 11 such decisions. In the same vein, our model needs four parameters, whereas, TBRS* needs no <10. Second, TBRS2 dynamics are more transparent than those of TBRS* because they rely on a mere analytical translation of the TBRS assumptions instead of a connectionist model. Thus, we can mathematically *prove* certain consequences of the TBRS assumptions, such as functional entanglement between the decay and refreshment functions.

In Sections 2-4, we will describe TBRS2 and prove certain theorems directly following from the TBRS assumptions. Six variants of TBRS2 will then be described that vary depending on how the refreshment process occurs. In the final section, we will use empirical data to illustrate how TBRS2 can be used in future research to test TBRS quantitatively and precisely.

[1] TBRS* is governed by the core theoretical assumptions of TBRS, but the authors also acknowledge that their model was based on more specific modelling decisions such as opting for positional coding to represent order (meaning that each item is associated to a position marker in TBRS*). This specific option for the model suffices to complexify the original model, and renders the model less handy. For instance, the authors specify that (1) position markers are predefined representations of serial positions, (2) neighbouring position markers are assumed to be similar to each other, (3) representations of positions were distributed using overlapping patterns of activation for which similarity decreased with distance, and (4) a random pattern was generated for the first position and subsequent patterns for next positions were derived by changing a random subset of the precedent pattern. Also, all processes (e.g., refreshing, processing of distractors) were governed exponentially over time, and time spent on a given process was finished when a latent variable had reached a criterion, based on accumulator models of response time, and processing rates were drawn from a Gaussian distribution. The authors specify other implementations for encoding (items are associated to their position by hebbian learning), item errors (the model was augmented by introducing a threshold for retrieval as a new parameter to account for omissions), response suppression (response suppression was implemented by hebbian anti-learning), etc. Therefore, the model is far more complex, even if similar parameters are present such as processing rate R, decay rate D, refreshment duration, τ. However, in contrast to TBRS2, TBRS* does not require a baseline activation because the encoding of an item and its association to a position are governed by hebbian learning, starting from zero. However, a crucial difference is that TBRS* uses a baseline owing to response suppression, where the strength of an item is equal to zero, whereas TBRS2 does not require such a limit to predict recall probability.

2. Overview

Figure 1 provides a general overview of TBRS2. First, a complex span task (as described above) can be modelled by a 'task function', that is, a function of time indicating whether a memorandum is presented, a processing task is performed, or neither of these events occurs, at time t. When an item is presented, attention focuses on the item (according to TBRS). When a processing task is performed, the attentional focus is drawn away from memoranda. The remaining 'spare' time is dedicated entirely to refreshing items. How refreshment of the items is spread along the timeline depends on a strategy that TBRS leaves unspecified. Once a decision is made about the refreshment schedule, we can derive a 'focus function' from the task function. The focus function is a function of time indicating attentional focus (towards processing or encoding/refreshing an item). The graphical example given in Figure 1 is based on the assumption that each item is refreshed for a fixed duration, starting anew from the first item after each interruption.

Once the focus function is set, we can derive the activation dynamics of each item. Here, *activation* stands for the odds of correct recall of an item at a given time, that is, the odds that the item would be recalled if the participants had to recall it at time t. The focus function translates into the activation dynamics through the decay and refreshment functions: when attentional focus spots item x_i, the activation of x_i increases by the effect of refreshment. Otherwise, the activation decreases by the effect of time. As we will demonstrate below, the TBRS assumptions impose a direct link between the decay and

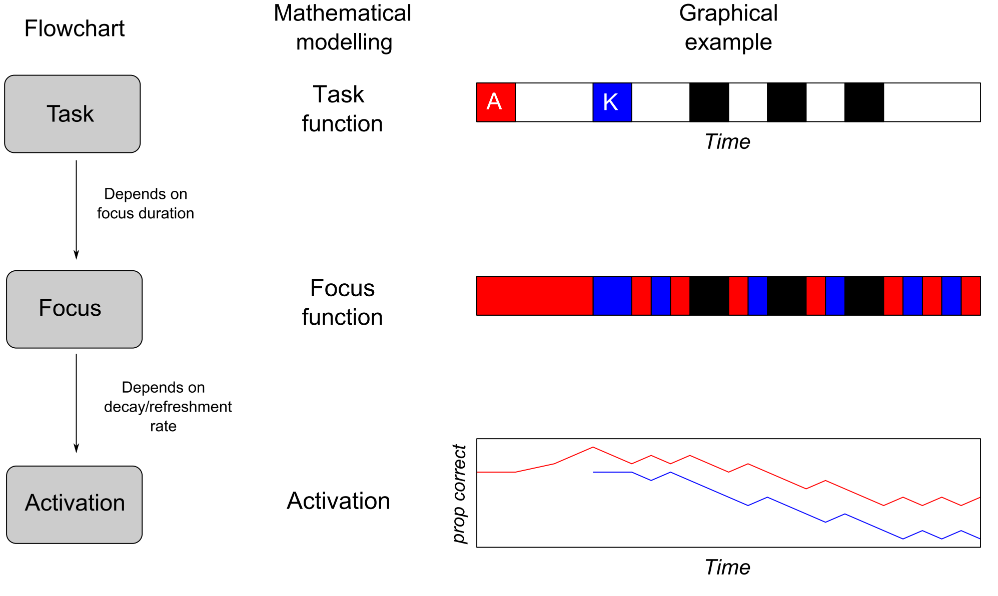

Figure 1. Overview of TBRS2's main functions. Complex span tasks are modelled by a task function indicating what is happening along the timeline (presentation of a new memorandum [A or K], free time [white], or processing tasks [black]). From the task function, we can derive a focus function indicating how attentional focus switches from items (colour) to processing distractors (black). To translate the task into focus, we must specify how spare time is used (e.g., how long each refreshment period lasts). Activation is the odds of correct recall for a given item and can be derived from the focus function as soon as the decay and refreshment rates are set. [Colour figure can be viewed at wileyonlinelibrary.com]

refreshment functions. In the following, we will present further details about the model, starting from the link between the focus function and activation, and then turning to the task functions and their relations with the focus functions.

3. From focus function to activation

Let us consider a situation in which a sequence of items ($x_1, \ldots, x_k$) is to be memorized within a period of time $[0, T]$. At any instant $t \in [0, T]$ after its presentation, item x_i has an *activation* $a_i(t)$, here defined as the odds that x_i would be correctly recalled at time t by the participant, who would be required to recall items at this point in time.

3.1. The dynamics of activation

We later assume that the decay of memory traces (i.e., activation) is exponential, following previous studies (Bolhuis, Bijlsma, & Ansmink, 1986; Wickelgren, 1970; Zylberberg, Dehaene, Mindlin, & Sigman, 2009). However, we first consider a more general case of decay to prove a general link between the decay and refreshment rates that follows from the TBRS hypotheses. When attention is focused towards item x_i, the corresponding activation increases in such a way that a_i is a solution of an autonomous differential equation $y' = R(y)$. The refreshment function R is continuously positive and does not depend on i. The previous equation only formalizes the idea that the refreshment rate does not depend on time *per se*, but on the current activation of the item. Likewise, when attention is drawn away from x_i, a_i decreases following an autonomous differential equation $y' = -D(y)$, where D is a continuous positive function independent of i.

Two remarks should be made here. First, the decay and refreshment of an item activation are independent of the activation of other items. This ensues from the TBRS assumption that temporal factors are preponderant. Second, we do not define an activation threshold under which the memory trace is permanently lost, such that there is no true forgetting. This counter-intuitive phenomenon is the consequence of the TBRS assumption that the probability of recall is a function of the cognitive load.[2] The absence of threshold (i.e., the absence of irremediable forgetting, theoretically) is psychologically implausible. However, one should keep in mind that this is a mathematical consequence of a strict application of the cognitive load assumption, not a psychological assumption; that even if the mathematical model contains such a feature, it might still be reliable in predicting recall in ecological situations; and that the theoretical possibility of retrieving any item in memory, however low its activation, does not imply an item can always be retrieved *in practice*.

3.2. Focus function and cognitive load

Let us first define a function describing the dynamics of attentional focus with respect to an item x_i.

[2] Imagine a task in which one has to memorize a single item, with an alternation of free time and task processing each second. The item can be kept in memory as long as needed, say 60 s. Then, imagine another task in which the participant is continuously distracted for 30 s, and then has 30 s of spare time. Because of the cognitive load assumption, the probability of recall should be the same in both examples, the cognitive load being 50% in both cases. Thus, the item is not lost in 30 s (or, in fact, in any duration).

Definition The focus function φ_i of item x_i is defined as $\varphi_i(t) = 1$ if the item is being refreshed at time t (i.e., attention is focused on x_i), $\varphi_i(t) = 2$ if item x_i is displayed at time t, and $\varphi_i(t) = 0$ otherwise.[3]

Definition The item cognitive load associated with item x_i on time interval $[t_0, t_1]$ is defined as the proportion of time not devoted to the item, that is,

$$CL_i(t_0, t_1) = \frac{\mu\{[t_0, t_1] \cap \varphi^{-1}(0)\}}{t_1 - t_0}, \tag{1}$$

where μ is the usual Borel measure and $\varphi^{-1}(0)$ is the set of all instants t such that $\varphi(t) = 0$. Thus, $\mu\{[t_0, t_1] \cap \varphi^{-1}(0)\}$ is the amount of time not devoted to refreshing item x_i between t_0 and t_1.

Note that, when computing this cognitive load, we will always assume that item x_i was presented before t_0 (and thus does not appear during $[t_0, t_1]$).

The TBRS model's core assumption is that $a_i(t_1)$ depends only on $CL_i(t_0, t_1)$ and its initial value $a_i(t_0)$. From this *cognitive load assumption*, a relation between D and R can be derived.

Theorem 1. *Under the cognitive load assumption, R and D are proportional: $R = \kappa D$, $\kappa \in \mathbb{R}_+^*$.*

Proof. Consider a situation in which a unique item (x) is to be remembered at time T, and such that attention is drawn away from x except on an interval $[\tau, \tau + h]$ (think of h as 'small').

The activation $a(t)$ is thus decreasing on $[0, \tau]$ and $[\tau + h, T]$, but increasing on $[\tau, \tau + h]$. The cognitive load at T does not depend on $\tau \in [0, T - h]$, thus $a(T)$ does not depend on τ either.

There exists a single $e > 0$, such that $a(\tau + h + \epsilon) = a(\tau)$, where e is the time needed for a to go back down to the level it was at τ, before refreshment. The cognitive load assumption implies that $h+\varepsilon$ is independent of either τ or $y = a(\tau)$.

When $h \to 0$, so does ϵ. Let $\delta = a(\tau + h) - a(\tau)$. Considering $a(t)$ on $[\tau, \tau + h]$, we find

$$\frac{\delta}{h} \longrightarrow R(y).$$

Considering $a(t)$ on $[\tau + h, \tau + h + \epsilon]$, we have

$$\frac{\delta}{\epsilon} \longrightarrow D(y);$$

thus,

$$\frac{\epsilon}{h} \longrightarrow \frac{R(y)}{D(y)}.$$

Because h and ϵ are independent of y if the cognitive load assumption is satisfied, we must have $R = \kappa D$, $\kappa = \lim(\epsilon/h) \in \mathbb{R}_+^*$. □

[3] Note that $\varphi_i(t) = 2$ is only dummy code ascribed to a particular event.

Thus, D and R are proportional under the cognitive load assumption expressed in the TBRS model. This is a mathematical consequence of a main TBRS hypothesis that has never been expressed.

3.3. Exponential decay

For the sake of simplicity, we suppose that whenever an item is first presented, its activation equals a constant baseline value β during presentation (this does not impair the generality of TBRS2, provided that the presentation duration is constant across memoranda).

Henceforth, we will also assume that the decrease in activation is exponential, which amounts to saying that D is a linear function. From Theorem 1, we know that R is then also a linear function (i.e., the refreshment is exponential). In other words, if attention is not focused on item x_i, then $a_i(t) \propto \exp(-dt)$, where d is the (absolute) *decay rate*. If attention is focused on x_i, then $a_i(t) \propto \exp(rt)$, where r is the *refreshment rate*. For exponential decay, an easy way to study the probability of a correct recall is to consider log-odds instead of activation levels (odds). Indeed, if activation decay is exponential, then log-odds evolution is linear, with slope r and $-d$; hence, the following theorem (the proof is immediate).

Theorem 2. *Suppose that at time t_0, an item x_1 has activation $a_1(t_0)$. Let φ_1 be its focus function. If the item is never presented during period $[t_0, t]$, then*

$$\log(a_1(t)) = \log(a_1(t_0)) - d(t - t_0) + (d + r)\int_{t_0}^{t} \phi_1(u)du. \quad (2)$$

Figure 2 displays two simple examples of activation dynamics. The plots were made using the tbrs function in R[4] (R Core Team, 2016). Alternatively, one can use our user-friendly online `Shiny` application.[5]

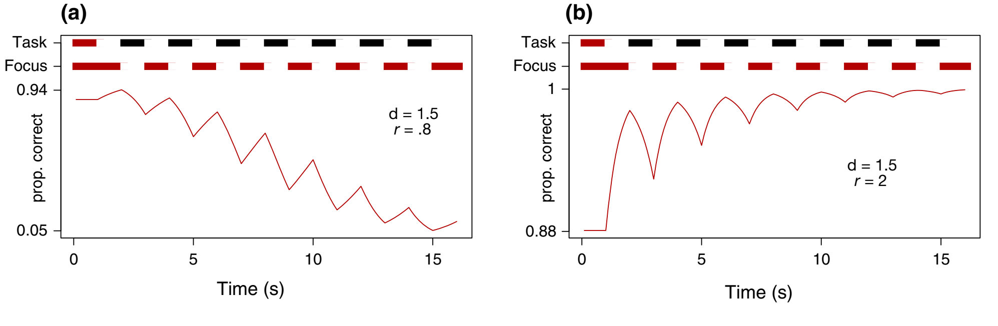

Figure 2. Examples of activation dynamics predicted by the model, with a focus on attention switching on and off of the memory item every second. During the first second, the item is presented and log-activation is set to 2. Two different sets of conditions (d, r) are presented. The memory trace fades away when (a) $d < r$, but not when (b) $d > r$. [Colour figure can be viewed at wileyonlinelibrary.com]

[4] Available at `https://github.com/ngauvrit/tbrs.git`
[5] `https://mathematicalpsychology.shinyapps.io/tbrs`

4. Task function and task cognitive load

So far, we have considered the case of a single memorandum. However, TBRS was designed to predict item recall in more complex tasks in which several items $(x_1, \ldots, x_n)$ are to be remembered.

Definition Define a task function as a function T of time, with T(t) = −1 if the concurrent task is being performed at t; T(t) = 0 if no concurrent task is to be performed at t with no item presented; and T(t) = k, where k ∈ {1, . . . , n}, if item x_k is being presented at time t.

For such a task, the *cognitive load* is the proportion of time exclusively devoted to the concurrent task on a given interval,

$$CL(t_0, t_1) = \frac{\mu\{[t_0, t_1] \cap T^{-1}(-1)\}}{t_1 - t_0}, \tag{3}$$

where μ is the usual Borel measure and $T^{-1}(-1)$ is the set of all *t* such that *T*(*t*) = −1. Thus, $\mu\{[t_0, t_1] \cap T^{-1}(-1)\}$ is the amount of time not devoted to the concurrent task, which means that this amount of time is devoted to refreshing the memoranda.

4.1. An invariant

Consider a task function *T* and time interval $[t_0, t_1]$ on which no item is presented. Part of this time $[CL(t_0, t_1)]$ is dedicated to the concurrent task, but the rest is devoted to the refreshment. Because the refreshment may be distributed among the different items in various specific time courses, the activation at time t_1 might vary. Consider, for instance, the case of two items *x* and *y*. If the refreshment is dedicated mainly to *x* during spare time, then we expect a high probability of recall for *x* and a low one for *y*; however, the reverse case is to be expected if the refreshment is dedicated mainly to *y*. Thus, how 'spare time' is distributed among the items is an important question for making quantitative predictions. However, because every brief pause is assumed to be dedicated to attentional refreshment, the product of activations is an invariant:

Theorem 3. *If no new item is presented on an interval* $[t_0, t_1]$*, and n is the number at time* t_0 *of items to be remembered, then*

$$\prod_{i=1}^{n} a_i(t_1)$$

does not depend on how the refreshment time is distributed on $[t_0, t_1]$*; it depends only on* $\prod a_i(t_0)$ *and* $CL(t_0, t_1)$*.*

Proof. Consider log (a_i). On any interval dedicated to a dual task, a_i is decreasing, with slope $-d$, for all *i*. Thus, the sum

$$S(t) = \sum_{i=1}^{n} \log(a_i(t)) \tag{4}$$

is a linear function with slope $-nd$. On any interval on which attention is focused on an item, S(t) is also linear, with slope $r-(n-1)d$. Therefore, we have

$$S(t_1) - S(t_0) = -nd(t_1 - t_0)CL(t_0, t_1) + (r - (n-1)d)(t_1 - t_0)(1 - CL(t_0, t_1)). \tag{5}$$

Considering the exponential completes the proof. □

4.2. Simple span

The parameters r and d are directly related to simple span (memory capacity), defined as the maximum number of items one can maintain in memory when no concurrent task is involved. More precisely, the theoretical simple span k can be computed using a simple formula, as shown by Theorem 4. Using this formula, we can estimate a participant's simple span from any data gathered through a complex span task, and thus again test the TBRS assumptions.

Theorem 4. *Let k be the simple span (memory capacity) corresponding to a set of parameters. We have*

$$k = \left\lfloor 1 + \frac{r}{d} \right\rfloor,$$

where $\lfloor x \rfloor$ is the floor function (i.e., the largest integer smaller than or equal to x).

Proof. Consider a simple span task, in which n items are presented sequentially beginning at $t = 0$, and involving no dual task. Let t_0 be a time at which the n items have been presented. Then $CL(t_0, t)$ remains null, and thus

$$S(t) = S(t_0) + (r - (n-1)d)(t - t_0), \tag{6}$$

where

$$S(t) = \sum_{i=1}^{n} \log(a_i(t)).$$

Thus, $S(t)$ tends to $\pm\infty$, depending on the sign of $r-(n-1)d$.

If $r-(n-1)d > 0$, or $n < 1 + \frac{r}{d}$, then $S(t)$ tends towards ∞, as does

$$\prod a_i(t) = \exp(S(t)).$$

If $n > 1 + \frac{r}{d}$, then S tends towards $-\infty$, and

$$\prod a_i(t) \longrightarrow 0.$$

We have thus proven that

$$\prod_{i=1}^{n} a_i(t) \underset{t\to\infty}{\longrightarrow} 0$$

if $n > 1 + \frac{r}{d}$, and

$$\prod_{i=1}^{n} a_i(t) \underset{t\to\infty}{\longrightarrow} \infty$$

if $n < 1 + \frac{r}{d}$.

If n items can be maintained in memory, then no activation tends towards 0, and thus $\prod_{i=1}^{n} a_i(t)$ does not tend towards 0, which means that $n \leq 1 + r/d$. Thus, $k = \lfloor 1 + \frac{r}{d} \rfloor$ is the maximum number of maintainable items (i.e., the simple span). □

4.3. Summary

We implemented the assumptions expressed in the TBRS model in a mathematical framework based on the following axioms:

1. Attention is always focused on a single item to be remembered or on a concurrent task.
2. Activation of item x_i increases when attention is focused towards x_i and decreases otherwise. The rate of decay/increase is a function of the current activation.
3. Activation of item x_i at time t is a function of the cognitive load on $[t_0, t]$, provided that item x_i is presented before time t_0.
4. Activation decreases exponentially.

From these axioms borrowed from TBRS (except for the exponential decay), we derived the following mathematical consequences:

1. The refreshment rate is exponential.
2. Given activations $(a_1(t_0), \ldots, a_n(t_0))$ at time t_0, and provided that no new item is presented after t_0, the product of the activations at time $t > t_0$ does not depend on how the refreshment time is distributed across the memory items.
3. The refreshment rate r, decay rate d, and simple span k (which is the maximal number of items that can be maintained in memory in a simple span task) are linked by the straightforward relation

$$k = \left\lfloor 1 + \frac{r}{d} \right\rfloor.$$

5. Variants of the TBRS model

A given task defined by a function T leads to a time-dependent focus vector $(\varphi_1(t), \ldots, \varphi_n(t))$.

The TBRS assumptions require that for any $i \in \{1, \ldots, n\}$,

1. $\varphi_i(t)$ is undefined, if item i has not yet been presented at t;
2. $\varphi_i(t) = 0$ (or is undefined), if $T(t) = -1$;
3. $\varphi_i(t) = 2$ (and $\varphi_{j\neq i} = 0$ or is undefined), if $T(t) = i$; and
4. $(\varphi_1(t), \ldots, \varphi_n(t))$ has exactly one component equal to 1, and all others are equal to 0 or undefined, if $T(t) = 0$.

The last point expresses that whenever no item is being presented, and when no concurrent task is required, attention is focused on one of the items to be remembered. However, it does not predict how spare time is distributed among items. We will now define six variants of the TBRS2 model based on how the spare time period is organized to deal with the memory items.

5.1. Steady versus threshold

A first distinction can be made regarding how long an item is refreshed when attention is focused on it. A variant of TBRS2 that we will call *steady* posits that the refreshment duration is a fixed value (for instance, $d = 0.2$ s). Another variant (the *threshold* model) posits that whenever attentional focus switches to a new item, it does so until activation of this particular item reaches a threshold w (unless attentional focus is directed away by a concurrent task).

Figure 3 shows examples of predicted activation dynamics in the case of a simple span task for the steady and threshold variants. In these examples, the steady variant predicts more variability in the final probability of recall than the threshold variant, if the number of items (here, three) is below the simple span. However, it predicts greater variability if the number of items (here, four) is above the simple span.

5.2. First, next, or lowest

The TBRS2 model can also vary in how it handles interruptions caused by concurrent tasks. During 'spare time', items are refreshed in a regular order: item 1, item 2, item 3, . . . ,

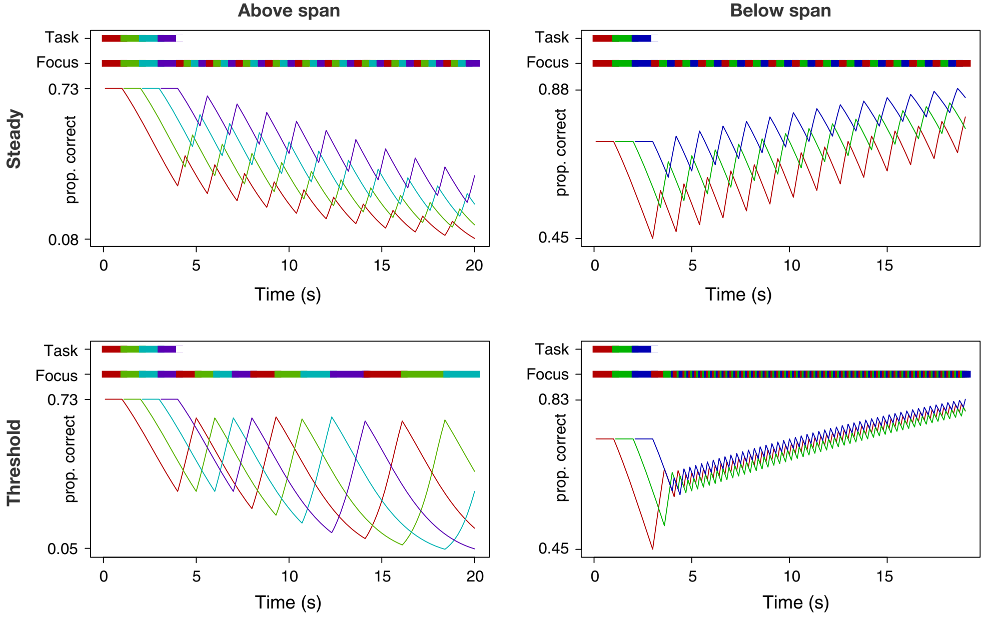

Figure 3. Variants of TBRS2 predictions for a simple span task. We set $d = 0.6$ and $r = 1.4$. In the steady variant, refreshment of an item lasts 0.4 s, whereas in the threshold variant, refreshment stops when activation reaches a cut-off point, or after 0.1 s (i.e., the refreshment lasts 0.1 s if the activation is already above the threshold at the beginning of the refreshment period). [Colour figure can be viewed at wileyonlinelibrary.com]

item n, item 1 . . .. However, there are different ways to select the item to be refreshed after an interruption is caused by the concurrent task. We will consider three simple variants. In the first variant, the first item x_1 is always refreshed first after the presentation of a distractor. In the next variant, the model keeps track of the last refreshed item and continues with the next one. For instance, if the concurrent task occurs while item 2 is refreshed, then item 3 will be refreshed after the interruption. Finally, the *lowest* variant predicts that the item with the lowest activation is refreshed first. This could correspond to a 'maximin' strategy in which one tries to maximize the minimal activation.

Depending on how dual task interruptions are spread in time, these variants may lead to different predictions that could hardly be presented by a verbal theory. A few illustrative examples are shown in Figure 4. With a particular task (alternation of a concurrent task and spare time every 2 s), visual inspection reveals different predictions depending on the variant. The first version predicts a primary effect, whereby the first item is more likely to be recalled. The next version predicts a recency effect, whereby the last item is more likely to be recalled. Finally, the lowest version predicts similar decreases in activation among items and no clear-cut order effect.

6. Parameter estimation and model comparison

Time-based resource sharing is underdefined when it comes to two characteristics. First, the question of refreshment duration remains unclear. TBRS assumes the refreshment

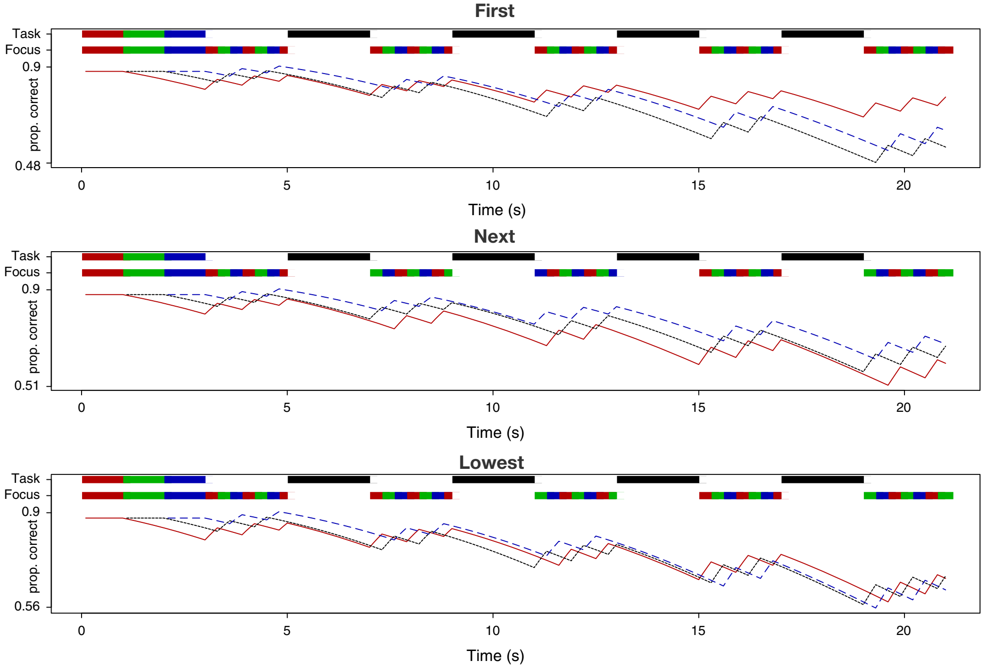

Figure 4. Predictions of the steady TBRS with three variants, with alternating distractors and spare time (2 s each). Parameters are set as follows: $d = 0.3$, $r = 1$, duration $= 0.3$ s, $\beta = \exp(2)$ (log-odds at presentation is set to 2). Here, we simplify a complex span task by putting the memory items at the beginning of the task. [Colour figure can be viewed at wileyonlinelibrary.com]

duration of an item to be such that the item will survive the next processing episode (Barrouillet & Camos, 2014). This is not directly usable in our model for two reasons. One reason is that there cannot be a formal threshold under which an item is totally lost in the model (still, we adapted this idea in the 'threshold' variant of the model). The other is the model should apply generally, that is, even in tests where processing episodes cannot be anticipated by participants (when a task, such as the one we run here, does not show regular patterns of switching between storage and concurrent task). Therefore, although the idea that participants can anticipate forthcoming processing episodes in order to better adapt refreshment strategies accordingly is psychologically plausible, it cannot be integrated unmodified in a more general version of the model, enabling unpredictable timelines. As mentioned above, we suggest two models concerning refreshment duration (steady and threshold).

Second, TBRS does not detail how the item to be refreshed is determined after an interruption. We thus consider three versions of the model regarding this point (first, steady and lowest). Combining these possibilities (two versions regarding refreshment duration and three versions regarding which item to refresh after an interruption), we construct six versions of TBRS2. In the following, we analyse experimental data using these variants. Note that all six variants have the same number of parameters (four), as follows:

1. decay rate d, expressed in points of log-odds per second;
2. refreshment rate r, also expressed in points of log-odds per second;
3. baseline β, which is the activation of an item when presented; and
4. duration (of the focus of a particular item) or threshold w.

In this section, we use empirical data to illustrate how the formal framework of TBRS2 can be used to compare models, estimate parameters, and gauge TBRS assumptions. The following analyses are included here for illustrative purposes, as an example showing how a particular example of data relevant to TBRS can be used and analysed.

6.1. Method

Thirty-two psychology students aged 18-29 years (M = 20.83, SD = 2.58) were recruited to take part in the experiment for course credit. Each participant performed a series of complex memory span task trials. In contrast to previous experiments in which the dual task is regular, the present dual task was semi-randomly organized along the timeline to test TBRS using the most diversified patterns of distraction.

6.1.1 Procedure

The stimuli were capital letters (B, F, H, J, K, L, P, Q, R, S, V, X) chosen for having few phonological similarities in French. The stimuli were displayed visually on a computer screen. Each list was composed of a maximum of six letters that were drawn without replacement. After each letter, a concurrent task required the participant to press the space bar whenever a 1, 2, or 3 stimulus digit was displayed. The stimulus digits were drawn randomly from the 1–9 set. The concurrent task occurred during a free time duration that was randomly drawn between 1 and 5 s ('free time' indicates only that the participant was not presented with the stimulus letters to be recalled, but does not imply that they were free enough to refresh the letters, as explained below). This free time was divided into 1,000 ms time slots during which attention capture could occur. For each slot, there was a chance for a distractor to be presented.

Each experimental session lasted approximately half an hour and included 60 separate stimulus lists. The 60 lists were built as follows: the length varied from two to six letters, and difficulty varied from easy to difficult. There were six different difficulty conditions for the concurrent task, based on six different probabilities (0, .20, .40, .60, .80, 1) that one stimulus digit would be drawn during each slot of the free time period. Once a probability was set for a list, it was applied to the entire list across the slots. This generated 5 (length) $\times$ 6 (difficulty) = 30 conditions, which we doubled to give each participant 60 lists to be recalled. For instance, if a 5 s free time duration, divided into five slots of 1,000 ms each, was chosen between two letters for a given list, and the probability was set to .80, the probability that one digit could be displayed in each of the five time slots was .80. This could, for instance, generate a 90,827 sequence (with the '0' symbol indicating a period of 1,000 ms without any distractor). The letters and the digits to be captured (1, 2, or 3) were always followed by an 'empty' slot to avoid building sequences that would be too cluttered. An example of a sequence is K01050V02087X000Q980. The participants could enter their response by clicking on a visual keyboard of 3 $\times$ 4 letters. The letters were always associated with the same positions on the keyboard across trials. The letter disappeared after being clicked, so the participants were not able to correct their response. The subjects were instructed to recall the letters in order, if possible. After the participants validated their answer with the space bar, a feedback screen indicated whether the recall was correct (i.e., item memory and order memory both correct). Then a screen with a GO window waited until the user moved on to the following list by pressing the space bar again. The participants were then presented with the next list, which followed a fixation cross lasting 2 s. The task began after a short warm-up including 18 progressive conditions. For the warm-up only, a rapid green light appeared on the screen under the digit location whenever a digit was correctly captured. Similarly, a rapid red light appeared whenever the space bar was pressed in error (false alarm). After the warm-up, the experimenter checked whether both tasks (memory task and concurrent task) were correctly performed during the warm-up before running the actual experiment. Omissions, false alarms, and recall performance were scrutinized by the experimenter to give the best advice to the participant for the following experiment (e.g., to be less impulsive on the concurrent task or to be more attentive to the concurrent/memory task).

6.2. Results

For this first simulation, we chose to analyse the data without taking the order or success of the concurrent task into account; a letter item was considered correctly recalled whenever it appeared in the response. For each participant and variant of TBRS2, a non-linear minimization of $-LL$ (where LL is the log-likelihood of the observed data) based on Newton's algorithm was performed. The constraints imposed on d and r were that $r > 0$ and $2 < r/d < 11$.[6]

To get a baseline, we computed the log-likelihood of a dummy model assigning equal and constant probability of recall to every item. For instance, if a participant recalled 95% of the 240 memoranda, the dummy model predicted a probability of recall equal to .95 for

[6] This restriction implies that the estimated simple span lies between 3 and 11, a range encompassing the range of simple spans usually reported.

every item in every trial. Note that the dummy model, although simplistic, gives an almost perfect fit for a proportion of correct recall nearing 1.

The overall proportion of items correctly recalled across all trials and subjects is high (94.2%). However, the proportion of trials in which all the memorandum is correctly recalled decreases as a function of the number of items to be recalled, with 98.3%, 94.0%, 88.9%, 75.9% and 63.4% respectively for 2, 3, 4, 5, and 6 items to be recalled. These data correspond to an overall complex span (i.e., the number of items that can be memorized in a complex span task) estimate equal to 4.2, computed through the formula $\sum ip_i$, where i is the number of items to be recalled in a trial, and p_i the proportion of correct responses in such trials.

The maximum likelihood estimate of the parameters was computed for each participant and variant of TBRS2. Table 1 displays the mean and standard deviation of these estimates, as well as an overall estimate of the simple span for each variant. The results concerning the log-likelihood are displayed in Table 2. Details on log-likelihood computation are given in the Appendix.

TBRS2 clearly fits the data better than the dummy model in terms of *LL* (Table 2). However, TBRS2 has four free parameters, whereas the dummy model has only one free parameter. We thus used the Akaike information criterion (AIC) to compare the models. The results are given in Figure 6, in which the area below the dotted horizontal line corresponds to cases for which TBRS2 yields a poorer fit than the dummy model in terms of the AIC.

From the estimated parameters d and r, we could derive an estimate of the simple span using Theorem 2. Because of constraints imposed on r/d, this estimated simple span was bound to lie between 3 and 12. The mean estimated simple span was 7.09 (SD = 1.53), with a median equal to 8. Figure 5 displays a violin plot of the participantwise estimated $1 + r/d$.

6.3. Discussion

As illustrated in Figure 6, the TBRS2 variants gave a better fit (AIC) than a dummy model whenever a participant correctly recalled <94% of the items. Keeping that in mind, the data are clearly in favour of TBRS2, as compared with the dummy model.

Comparisons of non-embedded models based on the AIC are always subjective; however, some authors have suggested that a difference of 4–7 (i.e., a difference of 2–3.5 in terms of *LL* for TBRS2) roughly corresponds to a 95% confidence interval (Burnham &

Table 1. Mean (*SD*) of the participantwise estimates of the parameters, sorted by variants of TBRS2

Variant	d	r	Duration	Baseline	Simple span
SF	0.098 (0.072)	0.62 (0.48)	0.42 (0.17)	3.85 (3.44)	7
SL	0.099 (0.07)	0.63 (0.46)	0.41 (0.16)	3.77 (3.42)	7
SN	0.097 (0.071)	0.62 (0.47)	0.40 (0.16)	3.84 (3.43)	7
TF	0.088 (0.075)	0.53 (0.48)	0.35 (0.15)	3.84 (3.46)	7
TL	0.085 (0.095)	0.50 (0.62)	0.35 (0.17)	5.10 (11.35)	6
TN	0.053 (0.24)	0.31 (1.57)	0.34 (0.16)	5.49 (13.36)	6

Note. SF = steady first; SN = steady next; SL = steady lowest; TF = threshold first; TN = threshold next; TL = threshold lowest. The last column displays an estimate of the simple span, defined as $\lfloor 1 + r/d \rfloor$.

Table 2. Participants' maximum log-likelihood table, sorted by increasing proportion of correct recall (column 1). The second column indicates the (minimum) negative log-likelihood of the dummy model. Columns 3–8 indicate the maximum log-likelihood difference between the dummy model and variant of TBRS2. Positive values indicate that a variant of TBRS2 fits the data better than the dummy model

Correct	Dummy	SF	SN	SL	TF	TN	TL
.79	124.14	13.90	14.36	13.81	11.82	11.88	11.46
.80	118.70	11.50	11.35	13.82	12.53	12.43	16.57
.81	115.82	9.41	9.41	10.60	11.25	12.61	13.15
.82	117.65	2.12	2.11	2.21	0.04	2.11	3.26
.85	101.45	10.58	9.46	9.27	13.63	15.56	13.33
.88	90.42	15.03	14.14	13.94	13.61	14.60	11.95
.88	86.46	8.98	9.29	7.98	9.36	12.16	9.58
.91	71.21	8.87	9.14	8.74	8.40	8.57	7.75
.92	66.42	3.84	3.94	2.66	3.02	3.11	3.11
.93	61.39	7.70	7.33	6.16	5.86	6.52	3.97
.93	58.78	1.77	1.40	1.47	1.54	1.49	1.57
.94	56.11	2.14	2.18	1.71	2.21	3.24	2.88
.94	56.11	5.86	7.14	6.17	5.98	6.41	6.08
.94	56.24	0.46	0.47	0.67	0.38	0.38	1.20
.95	50.55	1.16	1.14	0.95	1.42	1.44	0.87
.95	50.55	3.65	3.48	2.22	3.30	3.72	2.92
.95	44.65	12.57	12.70	14.41	13.69	13.82	14.92
.96	41.57	4.79	3.83	2.59	5.28	5.18	3.26
.96	38.38	1.43	1.44	2.25	1.75	1.91	2.55
.96	38.38	0.78	0.68	0.77	1.48	1.19	0.15
.96	38.38	0.67	0.70	0.58	1.21	1.75	0.04
.97	31.64	4.54	4.61	4.27	4.78	4.37	5.43
.97	28.06	2.29	2.38	1.90	2.09	1.89	2.21
.98	24.30	1.52	1.22	0.99	0.93	1.36	0.00
.98	20.34	3.23	3.88	3.05	4.61	2.58	2.34
.98	20.34	0.37	0.40	0.60	0.31	0.40	0.84
.98	20.34	3.45	3.00	3.09	3.62	3.58	2.48
.99	16.13	0.69	0.70	0.57	0.36	0.37	0.42
.99	16.13	0.35	0.36	0.32	0.56	0.60	0.13
.99	11.57	0.67	0.68	0.78	0.51	0.51	0.74
.99	11.57	0.23	0.23	0.19	0.20	0.20	0.23
.99	11.57	0.39	0.48	0.38	0.20	0.35	0.36

Note. SF = steady first; SN = steady next; SL = steady lowest; TF = threshold first; TN = threshold next; TL = threshold lowest.

Anderson, 2002). Using this rule of thumb, we found no strong evidence in favour of any variant of TBRS2 against another. There was no overall best variant across the participants, partly because several variants fitted different participants without apparent regularity. We defer the task of comparing variants of the TBRS2 to future research, as it is not the main goal of the current study.

Using Theorem 4, we derived participantwise estimates of the simple span and found that, although we imposed only loose constraints on this simple span, the resulting estimates are in line with previous research, suggesting a simple span of 5–9 (Mathy & Feldman, 2012; Miller, 1956). In fact, only two participants did not fall within this range.

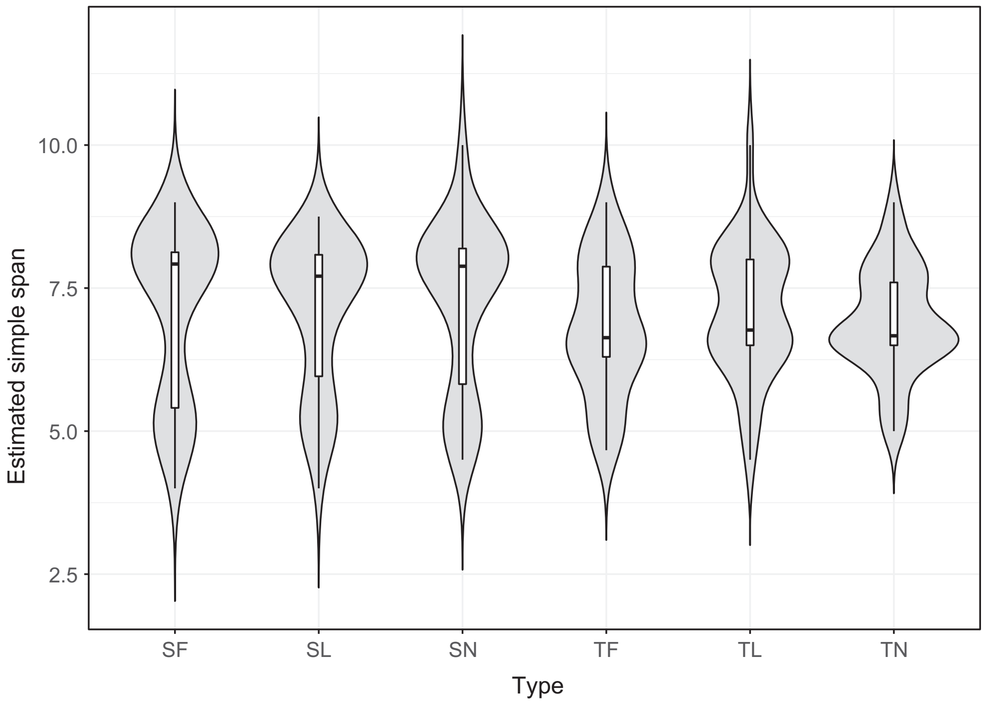

Figure 5. Violin plot of $1 + r/d$ (estimated simple span) by variants of TBRS2.

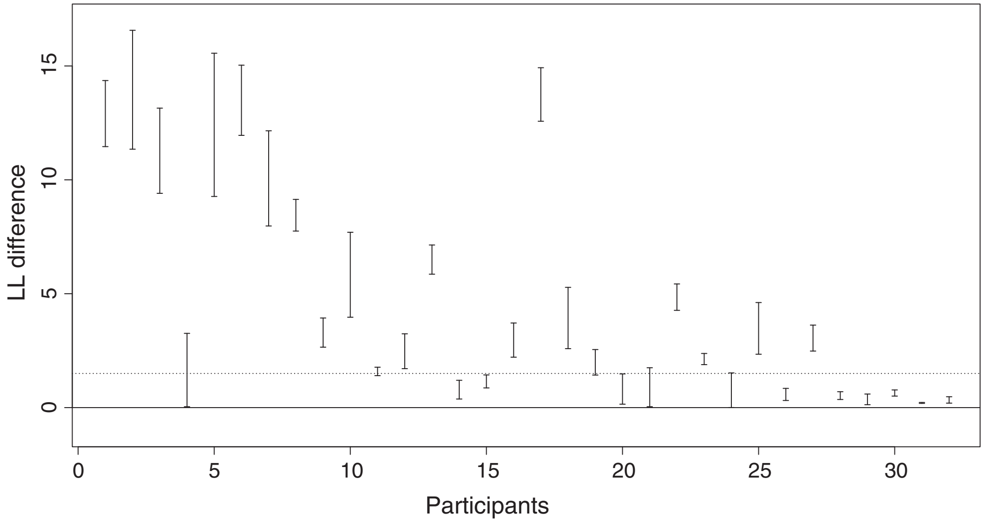

Figure 6. Maximum log-likelihood (*LL*) by participant, sorted by increasing proportion of correct recall. The *y*-axis displays the maximum log-likelihood difference between TBRS2 and the dummy model. Each segment runs from the minimum to the maximum value across the variants. Positive values indicate that TBRS2 fits the data better than the dummy model. The bottom area between the solid and dotted lines corresponds to cases in which, although the LL is greater for TBRS2, the Akaike information criterion is lower.

This result shows that the TBRS can provide plausible estimates of the average span. This illustrates one of the strengths of the mathematical transcript of TBRS proposed here: it allows an estimate of the simple span with data from which a direct measure of the simple span was not actually done since a concurrent task was used to limit the span (because the number of items to be remembered stays below the span). The participants could recall (complex span) 4.2 items on average, and never faced a trial in which they had more than 6 items to recall. However, we could derive estimates of d and r and thus, the theoretical simple span, from these data. This constitutes a method to rigorously test TBRS in the future, and illustrates one of the advantages of using a mathematical model. However, our data are not sufficient to reliably compare the variants of TBRS2 or even to settle the question of the relevance of TBRS2.

7. Conclusion

We constructed the first detailed mathematical transcription of TBRS assumptions that adds no characteristic not already addressed by the original description (Barrouillet & Camos, 2007), making as few decisions as possible and using as few parameters as possible. In comparison to the only other computational implementation, TBRS* (Oberauer & Lewandowsky, 2011), our TBRS2 model does not account for features such as order encoding: although it does predict order effects on correct recall, it does not describe how the order of the items is encoded. On the one hand, TBRS2 is thus less rich than TBRS*. On the other hand, TBRS2 is simpler and more transparent. Thanks to this transparency, we could prove several theorems mathematically following from the TBRS assumptions. For instance, the decay and refreshment functions are tightly related under the cognitive load assumption. Another striking theoretical result is that the simple span can be computed from the decay and refreshment rates. It is thus possible to indirectly estimate the simple span from data collected through complex span tasks using systematically fewer items to be recalled than the participants could virtually recall if no concurrent task was used. These results can now be used to test the TBRS theory at a degree of precision probably never reached before. In an illustrative experiment, we estimated the TBRS2 free parameters d and r and derived a simple span estimate (the simple span here being defined as the maximum number of items one can hold in memory for as long as needed). We found plausible results in favour of TBRS2 and therefore in favour of the TBRS theory.

Acknowledgements

The authors wish to thank Klaus Oberauer for providing helpful comments on a previous version of this work.

References

Baddeley, A. (2003). Working memory: Looking back and looking forward. *Nature Reviews Neuroscience*, *4*, 829–839. https://doi.org/10.1038/nrn1201

Barrouillet, P., Bernardin, S., & Camos, V. (2004). Time constraints and resource sharing in adults' working memory spans. *Journal of Experimental Psychology: General*, *133*(1), 83–100. https://doi.org/10.1037/0096-3445.133.1.83s

Barrouillet, P., & Camos, V. (2007). The time-based resource-sharing model of working memory. In N. Osaka, R. H. Logie & M. D'Esposito (Eds.), *The cognitive neuroscience of working memory* (pp. 59–80). Oxford: Oxford University Press.

Barrouillet, P., & Camos, V. (2012). As time goes by temporal constraints in working memory. *Current Directions in Psychological Science*, *21*, 413–419. https://doi.org/10.1177/0963721412459513

Barrouillet, P., & Camos, V. (2014). On the proper reading of the tbrs model: Reply to oberauer and lewandowsky (2014). *Frontiers in Psychology*, *5*, 1331. https://doi.org/10.3389/fpsyg.2014.01331

Barrouillet, P., Gavens, N., Vergauwe, E., Gaillard, V., & Camos, V. (2009). Working memory span development: A time-based resource-sharing model account. *Developmental Psychology*, *45*, 477–490. https://doi.org/10.1037/a0014615

Barrouillet, P., Portrat, S., & Camos, V. (2011). On the law relating processing to storage in working memory. *Psychological Review*, *118*, 175–192. https://doi.org/10.1037/a0022324

Bolhuis, J. J., Bijlsma, S., & Ansmink, P. (1986). Exponential decay of spatial memory of rats in a radial maze. *Behavioral and Neural Biology*, *46*(2), 115–122. https://doi.org/10.1016/s0163-1047(86)90584-4

Braver, T. S., & Bongiolatti, S. R. (2002). The role of frontopolar cortex in subgoal processing during working memory. *NeuroImage*, *15*, 523–536. https://doi.org/10.1006/nimg.2001.1019

Burnham, K. P., & Anderson, D. R. (2002). *Model selection and multimodel inference: A practical information-theoretic approach*. New York, NY: Springer Science & Business Media.

Camos, V., Mora, G., & Barrouillet, P. (2013). Phonological similarity effect in complex span task. *The Quarterly Journal of Experimental Psychology*, *66*, 1927–1950. https://doi.org/10.1080/17470218.2013.768275

Conway, A. R., Cowan, N., Bunting, M. F., Therriault, D. J., & Minkoff, S. R. (2002). A latent variable analysis of working memory capacity, short-term memory capacity, processing speed, and general fluid intelligence. *Intelligence*, *30*(2), 163–183. https://doi.org/10.1016/s0160-2896(01)00096-4

Conway, A. R., Kane, M. J., Bunting, M. F., Hambrick, D. Z., Wilhelm, O., & Engle, R. W. (2005). Working memory span tasks: A methodological review and users guide. *Psychonomic Bulletin & Review*, *12*, 769–786. https://doi.org/10.3758/bf03196772

Engle, R. W., Tuholski, S. W., Laughlin, J. E., & Conway, A. R. (1999). Working memory, short-term memory, and general fluid intelligence: A latent-variable approach. *Journal of Experimental Psychology: General*, *128*(3), 309–331. https://doi.org/10.1037/0096-3445.128.3.309

Farrell, S., & Lewandowsky, S. (2010). Computational models as aids to better reasoning in psychology. *Current Directions in Psychological Science*, *19*, 329–335. https://doi.org/10.1177/0963721410386677

Kane, M. J., & Engle, R. W. (2002). The role of prefrontal cortex in working-memory capacity, executive attention, and general fluid intelligence: An individual-differences perspective. *Psychonomic Bulletin & Review*, *9*, 637–671. https://doi.org/10.3758/bf03196323

Lewandowsky, S., Geiger, S. M., Morrell, D. B., & Oberauer, K. (2010). Turning simple span into complex span: Time for decay or interference from distractors? *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *36*, 958–978. https://doi.org/10.1037/a0019764

Mathy, F., & Feldman, J. (2012). Whats magic about magic numbers? Chunking and data compression in short-term memory. *Cognition*, *122*(3), 346–362. https://doi.org/10.1016/j.cognition.2011.11.003

Miller, G. A. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review*, *63*, 81–97. https://doi.org/10.1037/h0043158

Norris, D. (2005). How do computational models help us build better theories?. In A. Cutler (Ed.), *Twenty-first century psycholinguistics: Four cornerstones* (pp. 331–346). Mahwah, NJ: Lawrence Erlbaum Associates.

Oberauer, K., & Lewandowsky, S. (2011). Modeling working memory: A computational implementation of the time-based resource-sharing theory. *Psychonomic Bulletin & Review*, *18*(1), 10–45. https://doi.org/10.3758/s13423-010-0020-6

Pitt, M. A., & Myung, I. J. (2002). When a good fit can be bad. *Trends in Cognitive Sciences*, *6*, 421–425. https://doi.org/10.1016/s1364-6613(02)01964-2

Portrat, S., Barrouillet, P., & Camos, V. (2008). Time-related decay or interference-based forgetting in working memory? *Journal of Experimental Psychology: Learning, Memory, and cognition*, *34*, 1561–1564. https://doi.org/10.1037/a0013356

Portrat, S., Guida, A., Phénix, T., & Lemaire, B. (2016). Promoting the experimental dialogue between working memory and chunking: Behavioral data and simulation. *Memory & Cognition*, *44*, 420–434. https://doi.org/10.3758/s13421-015-0572-9

Pothos, E. M., & Wills, A. J. (2011). *Formal approaches in categorization*. Cambridge, UK: Cambridge University Press.

Prabhakaran, V., Narayanan, K., Zhao, Z., & Gabrieli, J. (2000). Integration of diverse information in working memory within the frontal lobe. *Nature neuroscience*, *3*(1), 85–90. https://doi.org/10.1038/71156

Team, R. C. (2016). *R: A language and environment for statistical computing*. Vienna, Austria: R Foundation for Statistical Computing, 2016. URL https://www.R-project.org/. ISBN3-900051-07-0.

Redick, T. S., Broadway, J. M., Meier, M. E., Kuriakose, P. S., Unsworth, N., Kane, M. J., & Engle, R. W. (2015). Measuring working memory capacity with automated complex span tasks. *European Journal of Psychological Assessment*, *28*, 164–171. https://doi.org/10.1027/1015-5759/a000123

Shahin, M. A., Jaksa, M. B., & Maier, H. R. (2009). Recent advances and future challenges for artificial neural systems in geotechnical engineering applications. *Advances in Artificial Neural Systems*, *2009*, 5. https://doi.org/10.1155/2009/308239

Swanson, H. L., & Beebe-Frankenberger, M. (2004). The relationship between working memory and mathematical problem solving in children at risk and not at risk for serious math difficulties. *Journal of Educational Psychology*, *96*(3), 471–491. https://doi.org/10.1037/0022-0663.96.3.471

Turner, M. L., & Engle, R. W. (1989). Is working memory capacity task dependent? *Journal of Memory and Language*, *28*, 127–154. https://doi.org/10.1016/0749-596x(89)90040-5

Unsworth, N., Redick, T. S., Heitz, R. P., Broadway, J. M., & Engle, R. W. (2009). Complex working memory span tasks and higher-order cognition: A latent-variable analysis of the relationship between processing and storage. *Memory*, *17*(6), 635–654. https://doi.org/10.1080/09658210902998047

van den Noort, M., Bosch, P., Haverkort, M., & Hugdahl, K. (2008). A standard computerized version of the reading span test in different languages. *European Journal of Psychological Assessment*, *24*(1), 35–42. https://doi.org/10.1027/1015-5759.24.1.35

Vergauwe, E., Barrouillet, P., & Camos, V. (2009). Visual and spatial working memory are not that dissociated after all: A time-based resource-sharing account. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *35*(4), 1012–1028. https://doi.org/10.1037/a0015859

Wickelgren, W. A. (1970). Time, interference, and rate of presentation in short-term recognition memory for items. *Journal of Mathematical Psychology*, *7*(2), 219–235. https://doi.org/10.1016/0022-2496(70)90045-3

Zylberberg, A., Dehaene, S., Mindlin, G. B., & Sigman, M. (2009). Neurophysiological bases of exponential sensory decay and top-down memory retrieval: A model. *Frontiers in Computational Neuroscience*, *3*, 4. https://doi.org/10.3389/neuro.10.004.2009

Received 8 September 2016; revised version received 26 June 2017

Appendix: Log-likelihood algorithm

The dummy model

For a given participant, the dummy model assumes that each item has the same probability p of being retrieved, p being the observed proportion of correct recall. Thus, the probability of an observation (the likelihood) is given by the formula $p^c+(1-p)^{n-c}$, where n is the total number of items to recall, and c the number of items correctly recalled. Thus, the log-likelihood is given by

$$LL = c\log(p) + (n-c)\log(1-p).$$

TBRS2 variants

Let us assume that a variant of TBRS2 is chosen, as well as a value for each parameter. For a given participant and task, each version of TBRS2 yields an estimate of the probability that each item will be correctly recalled at the end of the trial. Let us call these theoretical probabilities $p_1, p_2, \ldots, p_{\mathrm{n}}$ (n is the number of items to recall). At the end of the trial, we have a set of observations, $o_1, \ldots, o_n$ equal to 1 if the corresponding item is correctly recalled, and 0 otherwise.

The likelihood of the corresponding observation, that is, the probability that, within the TBRS2 model, such an observation arises, is given by the formula

$$\prod_i (o_i p_i + (1-o_i)(1-p_i)).$$

Note that in this formula, each $o_i p_i+(1-o_i)(1-p_i)$ is simply p_i if item number i is correctly recalled, and $1-p_i$ otherwise.

Thus, the log-likelihood corresponding to this task and participant is given by

$$\log\left(\prod_i (o_i p_i + (1-o_i)(1-p_i))\right) = \sum_i \log(o_i p_i + (1-o_i)(1-p_i)).$$

To compute the log-likelihood of a participant, we simply sum up the log-likelihoods corresponding to all trials from the participants.

Using a non-linear minimization method (R-function nlm), we could determine the maximum log-likelihood $\hat{L}$ corresponding to a participant. From this value, we derive the AIC using the formula

$$\mathrm{AIC} = -2\hat{L} + 8,$$

where eight is the value of $2k$, since $k = 4$ (the number of parameters in the model).

The R-scripts used to compute the log-likelihood are freely available online at `https://github.com/ngauvrit/tbrs`.