Acta Psychologica 144 (2013) 373–379

Contents lists available at ScienceDirect

Acta Psychologica

journal homepage: www.elsevier.com/locate/actpsy

Avoiding the problem state bottleneck by strategic use of the environment

Jelmer P. Borst [a,b,*], Trudy A. Buwalda [b], Hedderik van Rijn [b], Niels A. Taatgen [b]

[a] Carnegie Mellon University, Pittsburgh, PA, USA
[b] University of Groningen, Groningen, The Netherlands

ARTICLE INFO

Article history:
Received 21 January 2013
Received in revised form 18 July 2013
Accepted 22 July 2013
Available online xxxx

PsycINFO classification:
2300 Human Experimental Psychology
2340 Cognitive Processes
4010 Human Factors Engineering

Keywords:
Multitasking
Problem state bottleneck
Soft Constraints Hypothesis
ACT-R

ABSTRACT

We investigated whether environmental support can be used to circumvent the problem state bottleneck in human multitasking. Previously, it was shown that people can only maintain a single chunk of information in their problem state resource, the central part of working memory. Consequently, when the problem state resource was required by multiple tasks concurrently, performance decreased. This phenomenon was termed the problem state bottleneck. To investigate whether the environment can be used to circumvent this bottleneck, we conducted an experiment with two main conditions. In the No-Support condition we replicated an earlier experiment that indicated the existence of the problem state bottleneck. In the Support condition we presented external cues, reducing the load on the problem state resource. To support the results of the experiment we present a computational cognitive model. The experiment and model indicated that the problem state bottleneck can be avoided by using external cues. However, subjects only used external cues when this led to faster behavior. These results were interpreted in the light of the Soft Constraints Hypothesis, which states that humans always follow the fastest strategy possible, as opposed to the most accurate strategy.

1. Introduction

Multitasking is all around us. González and Mark (2004) have shown that people switch on average every 3 min between tasks in a typical office environment. In addition, a recent study indicated that every generation 'multitasks' more in their free time than the previous generation (Carrier, Cheever, Rosen, Benitez, & Chang, 2009). Although multitasking seems to become the norm in our societies, it is also well known that performance on individual tasks suffers from multitasking. In the field of sequential multitasking (i.e., switching between tasks, Salvucci, Taatgen, & Borst, 2009), theorists have focused on the disruptive effects of interruptions (e.g., Gillie & Broadbent, 1989; Monk, Trafton, & Boehm-Davis, 2008). Likewise, the concurrent multitasking literature has identified several processing bottlenecks that lead to decreased performance when two tasks are performed at the same time (e.g., Broadbent, 1958; Keele, 1973; Pashler, 1994; Salvucci & Taatgen, 2008, 2011; Wickens, 1984, 2002). One important cause of multitasking interference, both in concurrent and sequential multitasking, is the *problem state bottleneck* (Borst, Taatgen, & Van Rijn, 2010).

The problem state is defined as the element in working memory that can be used without a time cost (Anderson, 2005), unlike other elements in working memory (see e.g., McElree, 2001). It is used to represent intermediate information in a task, for example, '$3x = 15$' when solving '$3x - 5 = 10$'. Previously, it was shown that the problem state resource can contain at most one chunk of information, and therefore causes multitasking interference when required by multiple tasks at the same time (while using information in the problem state resource does not take time, storing a new chunk of information in the problem state resource was estimated to take 200 ms; Anderson, 2005; Borst, Taatgen, & Van Rijn, 2010). In a dual-task paradigm subjects needed a problem state for none, one, or both of the tasks. In the condition where subjects needed a problem state for both tasks, performance decreased considerably both in reaction times and accuracy as compared to the other conditions. Supported by a cognitive model, this was taken as an indication of a problem state bottleneck. Further evidence for a problem state bottleneck was provided by fMRI experiments (Borst, Taatgen, Stocco, & Van Rijn, 2010; Borst, Taatgen, & Van Rijn, 2011) and by Salvucci and Bogunovich (2010), who showed that when subjects had to switch between an e-mail and a chat task, they chose switch points at which they did not have to maintain a problem state.

Although the problem state bottleneck causes significant interference in experimental settings, in real life it is often possible to use the environment as an external memory (e.g., Hollan, Hutchins, & Kirsh, 2000; Kirsh, 1995; Wickens, 1992), thereby avoiding the limits of the problem state resource. For example, when solving a multicolumn subtraction problem on paper it is common to indicate whether a carry is in progress, which decreases problem state resource requirements. In cases where one relies on working memory, and stores the carry in the problem state

* Corresponding author at: Carnegie Mellon University, 5000 Forbes Ave., Pittsburgh, PA 15213, USA. Tel.: +1 412 268 6113.
E-mail address: jelmer@cmu.edu (J.P. Borst).

0001-6918/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.actpsy.2013.07.016

resource, there are two possible strategies to continue after losing the representation due to an interruption: recalling whether a carry was in progress from declarative memory, or reconstructing it by recalculating the previous column. While reconstruction is the safer option, it is also likely to take more time than recall from memory.

We hypothesize that these are strategic processes: when reconstructing or perceiving a problem state from the environment takes less time than a cognitive 'in-the-head' strategy (using the problem state resource as is, or retrieving information from memory), the environment will be used, otherwise a mental strategy will be applied. This is in accordance with the Soft Constraints Hypothesis, which proposes that our strategy choices aim at minimizing temporal costs instead of, for example, mental effort (Gray & Fu, 2004; Gray, Sims, Fu, & Schoelles, 2006). To support this, Gray et al. showed in four experiments that subjects always use the fastest local strategy, instead of minimizing mental effort or total time-on-task. Even when minimizing local time led to suboptimal behavior (using fast but imperfect knowledge in-the-head vs. slower but perfect knowledge in-the-world) or to more memory effort (memorizing multiple facts instead of perceptually revisiting a display), subjects opted for the fastest method (that is, the fastest *local* method, as this often relied on imperfect knowledge in-the-head it could lead to mistakes and longer total time-on-task).

In this paper we test this hypothesis with an experiment in which subjects were asked to perform two tasks concurrently. The first condition of the experiment aimed at replicating the basic problem state bottleneck interference effect. In the second condition we presented supporting information on the screen, thereby enabling the use of the environment as an external memory. According to the Soft Constraints Hypothesis, subjects should only use the information in the environment when this is faster than using the information in their heads. To support our experimental results we will present a computational model that incorporates the Soft Constraints Hypothesis, and show that it matches the results in the data.

2. Methods

In the experiment subjects had to alternate between a subtraction task and a text-entry task. Both tasks were presented in two versions: an easy version in which there was no need to maintain a problem state, and a hard version in which subjects had to maintain a problem state from one response to the next. In addition, in one condition external support was displayed on the screen for the subtraction task, enabling the use of the environment in the hard subtraction condition. Thus, the experiment had a 2 × 2 × 2 factorial within-subjects design (Subtraction Difficulty × Text-Entry Difficulty × Support).

2.1. Subjects

33 students of the University of Groningen participated in the experiment for course credit or monetary compensation of €10. Four subjects were rejected because they scored less than 75% correct where the other subjects scored >95% correct. Two subjects were rejected because they did not adhere to task instructions, and three because of recording problems of the eye tracker (eye-tracker results turned out to be uninformative with regard to the current question and are therefore not reported in this paper). This leaves 24 complete datasets (17 female, age range 18–43, mean age 20.5). All subjects had normal or corrected-to-normal visual acuity. Informed consent as approved by the Ethical Committee Psychology of the University of Groningen was obtained before testing.

2.2. Design

The subtraction task was shown on the left side of the screen, the text-entry task on the right (see Fig. 1). Subjects had to alternate between the two tasks: after entering a digit, the subtraction interface was disabled, forcing the subject to subsequently enter a letter. After entering a letter, the text-entry interface was disabled and the subtraction interface became available again.

The subtraction task is shown on the left side of Fig. 1. Subjects were asked to solve 10-column subtraction problems in standard right-to-left order. However, at each point in time, only a single column was visible. Although the problems were presented column-by-column, the subjects were instructed that the separate columns in a trial were part of a 10-column subtraction problem (in the practice phase subjects started out with a normal 10-column layout, only later they switched to solving the problems column by column). Subjects had to enter digits by clicking on the on-screen keypad with the mouse. In the easy, no problem state version, carrying was never needed because the upper digit was always larger or equal to the lower one. In contrast, the hard version required subjects to carry six times out of 10 possible columns. The assumption is that subjects use their problem state resource to store whether a carry is in progress.

The interface for the text-entry task is shown on the right in Fig. 1. Subjects had to enter 10-letter strings by clicking on the on-screen keyboard. In the easy version these strings were presented one letter at a time and subjects had to click the corresponding button on the keyboard. In the hard version, a 10-letter word was presented once at the start of a trial. Once a subject clicked on the first letter, the word disappeared and the remaining letters had to be entered one at a time, without feedback. After the initial presentation of the word in the hard condition, subjects could neither see what word they were entering, nor what they had already entered.

Because subjects had to alternate between the two tasks after every response, they had to keep track of whether a carry was in progress for the subtraction task and what the word was for the text-entry task while performing the other task.

In the support condition a marker on the screen indicated whether a carry was in progress in the subtraction task. Fig. 1 shows this condition. The '|' above the subtraction task indicates that currently no carry is in progress. The indicator turned into a '1' after a column that induced a carry (e.g., 3–4). Thus, in the Support condition for the subtraction task subjects could use the environment as an external memory.

2.3. Stimuli and apparatus

The stimuli for the subtraction task were generated anew for each subject. The subtraction problems in the hard version featured six carries, and resulted in 10-digit answers. The 10 letter words for the hard version of the text-entry task were handpicked from a list of high-frequency Dutch words (CELEX database; Baayen, Piepenbrock, & Van Rijn, 1993) to ensure that similarities between words were kept at a minimum. These stimuli were also used in the easy text-entry task, except that the letters within the words were scrambled (under the constraint that a letter never appeared twice in a row). Thus, subjects were presented pseudo-random sequences of letters that they had to enter one-by-one in the easy condition. By scrambling the words, we controlled for letter-based effects, while preventing the use of strategies to predict the next letter.

The experiment was presented full screen on a 20.1" monitor. Subjects were sitting at a normal viewing distance, approximately 70 cm from the screen.

2.4. Procedure

In this experiment, a trial was defined as the completion of the two tasks: solving one 10-column subtraction problem and entering one 10-letter word. Trials can be divided into 20 responses, 10 responses to the subtraction task and 10 responses to the text-entry task. Each trial started with the presentation of a fixation cross for 6 s, followed by two horizontally aligned colored circles representing the tasks. The

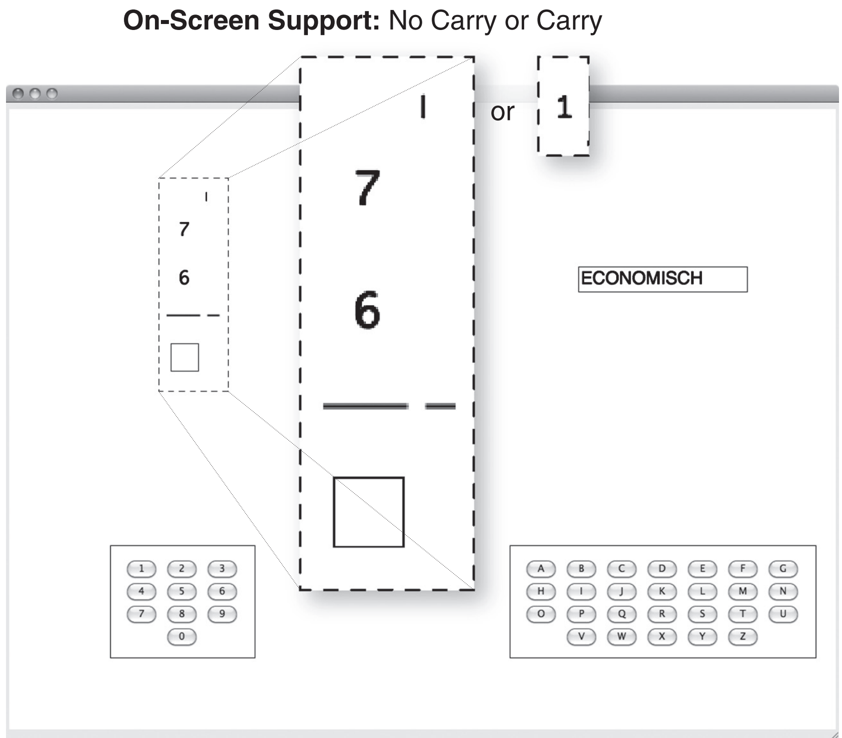

Fig. 1. The interface of the experiment in the Support condition. In the No-Support condition, nothing is shown above the subtraction column. The '|' in the subtraction task indicates that currently no carry is in progress, a '1' indicates a carry in progress. Note that in the real experiment only a single task was enabled at any given time.

color of the circles indicated the difficulty levels of the tasks (on the left for the subtraction task, on the right for the text-entry task; green for easy, red for hard). The circles stayed on the screen for 1 s, followed by a fixation cross for 600 ms, after which the subtraction and text-entry tasks appeared. Subjects had to begin with the subtraction task, and then alternated between the two tasks. After completing both tasks, a feedback screen was shown for 2 s, indicating how many letters/digits were entered correctly. Before the next trial started, a fixation screen was shown for 2 s.

The experiment consisted of two practice blocks and two experimental blocks. The first practice block consisted of 12 single task trials (4 subtraction trials with 10 columns visible, 4 subtraction trials with one column visible, and 4 text-entry trials). The second practice block consisted of 4 multitasking trials (all subtraction–text-entry conditions once: *easy subtraction–easy text-entry*, *hard subtraction–easy text-entry*, *easy subtraction–hard text-entry*, and *hard subtraction–hard text-entry*). Both experimental blocks consisted of 28 multitasking trials (7 trials of each condition, where each trial contained 10 responses to each task, see above). One of the experimental blocks contained the support condition; the order was counter-balanced over subjects. Before the second block the subtraction task was practiced again, to familiarize the subjects with using the carry indicator if they did not use this in the first block, or with performing the task without the indicator in the other case. Subtraction and text-entry conditions were randomized within a block. The complete experiment consisted of 56 experimental trials, and lasted for about 90 min. Between blocks subjects could take a short break.

2.5. Statistical procedure

Only data from the experimental phase of the experiment were analyzed. A response time in the subtraction task was defined as the time between a response in the text-entry task and a response in the subtraction task; a response time in the text-entry task as the time between a response in the subtraction task and a response in the text-entry task. First responses of each trial were removed. Then, extreme outliers were removed from the data (RTs < 250 ms or > 10,000 ms), as these anticipatory and very slow responses are unlikely to be related to the processing of the stimuli associated with this response. Next, we removed data exceeding three standard deviations from the mean per condition per subject (in total 2.2% of the data was removed). To analyze the data we applied linear mixed-effects models (LMEs), with subject as random effect (e.g., Baayen, Davidson, & Bates, 2008). For the accuracy data a binomial LME model was used. In all analyses, we used a binary coding, with the support condition, hard subtraction, and hard text-entry coded as 1. All error bars depict within-subject standard errors from the mean per condition (Morey, 2008).

3. Results

Fig. 2 shows the results of the experiment: the top panels show response times, the bottom panels accuracy. LME results are reported in Tables 1–3. Tables 1 and 2 show results on subsets of the dataset (respectively No-Support and Support), and Table 3 shows results on the full dataset. First, we discuss how the No-Support condition yields corroborating evidence for the problem state bottleneck concept. We then turn to the Support condition, to see how the effects of the problem state bottleneck change when external support is provided.

3.1. The problem state bottleneck

The No-Support condition of the current experiment is a replication of Experiment 1 in Borst, Taatgen, and Van Rijn (2010), which was the first in a series of three experiments that were used to argue in favor of a problem state bottleneck. The first and third columns of Fig. 2 show the response times and accuracy in the No-Support condition, for text-entry and subtraction, respectively. To investigate whether the results replicated our earlier study, we fitted an LME to the No-Support data, with Subtraction Difficulty and Text-Entry Difficulty as fixed effects, and subject as a random effect (Table 1).

With respect to the responses on the subtraction task, there was a large increase in response times with Subtraction Difficulty: when

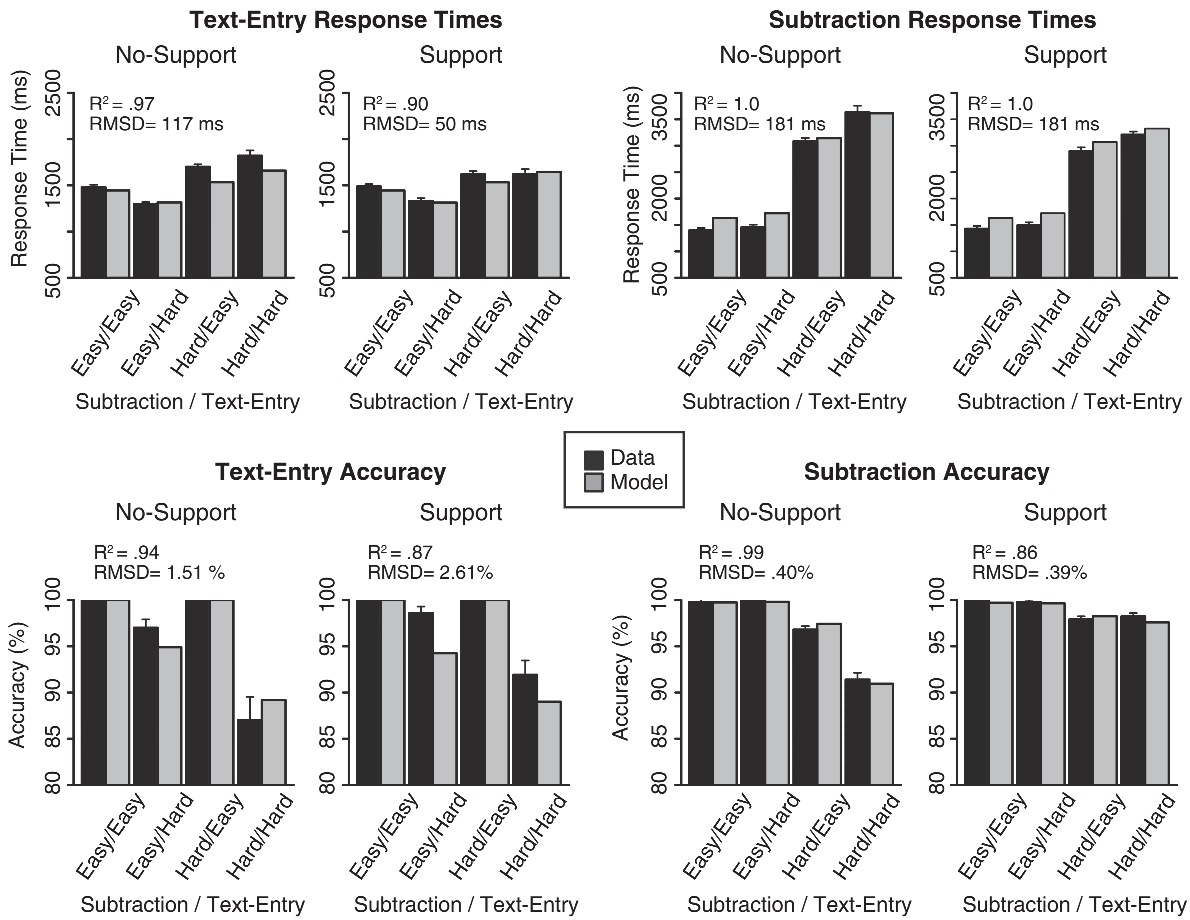

Fig. 2. Response times (above) and accuracy results (below) for the text-entry and subtraction tasks. The two columns on the left represent the text-entry task, the two columns on the right the subtraction task. The first and third columns depict the No-Support condition, whereas the second and fourth columns show the support condition. Error bars indicate within-subject standard errors from the mean per condition (Morey, 2008).

subtraction was hard, response times were higher than when subtraction was easy. More interestingly, when both tasks were hard, there was an additional increase in response time, as shown by a significant over-additive interaction effect between Subtraction Difficulty and Text-Entry Difficulty, with an estimated size of 483 ms. This interaction effect was taken as an indication of a problem state bottleneck: when subjects had to maintain a problem state for both tasks (*hard–hard* condition), response times increased considerably as compared to when they had to maintain a problem state for only one task (*easy–hard* and *hard–easy* conditions). Comparing this model to a model without the interaction term showed that the first model is to be preferred: $\chi^2(1) = 79.30$, $p < .001$, indicating that the added model complexity of the interaction term is warranted given the data.

A similar effect can be seen in the response times on the text-entry task. As long as *subtraction* was easy, text-entry responses were faster in the hard text-entry condition than in the easy text-entry condition (we discuss this effect in the model section below). However, response times increased when subtraction was hard, and responses were slowest in the *hard–hard* condition. Again, it seems that because an additional problem state is required in the other task, we see an increase in response times on the current task (the model below explains these effects in more detail). Statistically, this was shown by a significant interaction effect between Subtraction Difficulty and Text-Entry Difficulty.

Table 1
Linear mixed effects model results: No-Support. Betas indicate effects of when subtraction and text-entry were hard. Empty cells indicate factors that did not explain sufficient additional variance to be warranted.

	Response times			Accuracy		
Source	β	t	p	β	z	p
Subtraction task						
Intercept	1398.97	19.60	<.001	6.28	10.64	<.001
Subtraction (Hard)	1686.07	44.49	<.001	−2.79	−4.63	<.001
Text-Entry (Hard)	57.52	1.53	.13	14.36	.02	–
Subtraction × Text-Entry	482.90	8.93	<.001	−15.42	−.02	–
Text-Entry task						
Intercept	1479.23	42.37	<.001	21.72	.05	–
Subtraction (Hard)	221.29	12.25	<.001	−1.64	−9.38	<.001
Text-Entry (Hard)	−184.06	−10.10	<.001	−17.94	−.04	–
Subtraction × Text-Entry	282.4	10.79	<.001			

Table 2
Linear mixed effects model results: Support. Betas indicate effects of when subtraction and text-entry were hard. Empty cells indicate factors that did not explain sufficient additional variance to be warranted.

	Response times			Accuracy		
Source	β	t	p	β	z	p
Subtraction task						
Intercept	1432.54	23.80	< .001	6.93	12.34	< .001
Subtraction (Hard)	1466.86	45.76	< .001	−2.69	−5.07	< .001
Text-Entry (Hard)	65.89	2.06	.039	0.09	.35	–
Subtraction × Text-Entry	248.15	5.46	< .001			
Text-Entry task						
Intercept	1488.53	36.70	< .001	7.77	7.50	< .001
Subtraction (Hard)	131.71	7.25	< .001	13.23	.02	–
Text-Entry (Hard)	−159.70	−8.75	< .001	−3.08	−2.97	.003
Subtraction × Text-Entry	160.57	6.16	< .001	−15.11	−.02	–

Table 3
Linear mixed effects model results: overall. Betas indicate effects of when support was present, and when subtraction and text-entry were hard. Empty cells indicate factors that did not explain sufficient additional variance to be warranted.

	Response times			Accuracy		
Source	*β*	*t*	*p*	*β*	*z*	*p*
Subtraction task						
Intercept	1399.41	22.66	<.001	6.36	10.70	<.001
Support (On)	32.96	0.94	–	1.11	.95	–
Subtraction (Hard)	1686.07	47.63	<.001	−2.79	−4.65	<.001
Text-Entry (Hard)	56.40	1.61	.11	13.36	.03	–
Support × Subtraction	−220.18	−4.41	<.001	−0.67	−.56	–
Support × Text-Entry	9.51	0.19	–	−14.47	−.03	–
Subtraction × Text-Entry	482.65	9.56	<.001	−14.43	−.03	–
Support × Sub. × Text-Entry	−233.33	−3.29	.001	15.70	.03	–
Text-Entry Task						
Intercept	1479.50	43.60	<.001	9.75	9.56	<.001
Support (On)	9.09	.50	–	0.57	5.09	<.001
Subtraction (Hard)	221.06	12.04	<.001	−1.67	−11.96	<.001
Text-Entry (Hard)	−185.06	−9.99	<.001	−6.06	−6.02	<.001
Support × Subtraction	−89.30	−3.44	<.001			
Support × Text-Entry	25.08	.96	–			
Subtraction × Text-Entry	278.26	10.46	<.001			
Support × Sub. × Text-Entry	−116.69	−3.12	.002			

In addition, the LME with the interaction term was preferred to a model without the interaction term: $\chi^2(1) = 115.28$, $p < .001$.

The accuracy data of both tasks seem to show similar effects (Fig. 2, bottom panels, first and third column; LME results in Table 1). In case of the subtraction task, accuracy decreased when subtraction was hard, and it seemed to decrease further when text-entry was hard as well. While an LME with the interaction between Subtraction Difficulty and Text-Entry Difficulty explained sufficient additional variance to be warranted ($\chi^2(1) = 7.99$, $p = .005$), the interaction term in the model itself did not reach significance. The accuracy effects in the text-entry task followed the same pattern (bottom-left graph, Fig. 2). However, here model comparisons indicated that the interaction term was not warranted ($\chi^2(1) = 0$) – indicating an absence of an interaction effect. Note that all accuracy effects are somewhat hard to interpret given that the easy conditions are at ceiling.

Summarizing, response times on both tasks showed clear over-additive interaction effects for the *hard–hard* conditions, and the accuracy effects – although not significant – went in the same direction. Previously, these effects were taken as an indication of a problem state bottleneck (Borst, Taatgen, Stocco, et al., 2010; Borst, Taatgen, & Van Rijn, 2010). The current experiment replicates these earlier studies, lending additional support to the idea of a problem state bottleneck. The question is now whether and how these effects change when external support was provided.

3.2. External support: testing the soft constraints hypothesis

The second and fourth columns of Fig. 2 show the response times in the Support Condition for text-entry and subtraction, respectively. To test whether the problem state bottleneck could be circumvented, we investigated whether the two-way interaction between Subtraction Difficulty and Text-Entry Difficulty (indicating the bottleneck) decreased when support was provided. To this end, we fitted an LME model to all data (reported in Table 3). If the three-way interaction between Support, Subtraction Difficulty, and Text-Entry Difficulty reaches significance, this would indicate that the effects of the bottleneck are different for the Support and No-Support conditions. In addition to the full analysis, we also inspected separate models for Support (Table 2) and No-Support (Table 1) to see how the effects differed between those conditions.

With respect to the response times on the subtraction task, a significant three-way interaction between Support, Subtraction Difficulty, and Text-Entry Difficulty (Table 3) showed that the two-way interaction between Subtraction Difficulty and Text-Entry was smaller in the Support condition than in the No-Support condition. Thus, the difference between *hard–easy* and *hard–hard* condition was smaller with Support than without Support. However, as can be seen in Fig. 2 and Table 2, the two-way interaction between Subtraction Difficulty and Text-Entry Difficulty was still significant with Support: even with external support subjects showed an increase in response times in the *hard–hard* condition. The LMEs estimated the interaction effect without support to be 483 ms, and with support to be 248 ms (close to the three way interaction effect of −233 ms). Thus, the effect of the problem state bottleneck diminished, but did not disappear when external support was provided.

The response times on the text-entry task also showed a significant three-way interaction effect between Support, Subtraction Difficulty, and Text-Entry Difficulty. When external support was provided for the subtraction task, the effects of the problem state bottleneck decreased in the text-entry task. However, also here the two-way interaction effect between Subtraction Difficulty and Text-Entry Difficulty is still present with external support. The LMEs estimated the interaction effect without support to be 282 ms, and with support to be 161 ms.

The bottom panels of Fig. 2 show the accuracy data. The three-way interaction between Support, Subtraction Difficulty, and Text Entry Difficulty did not reach significance for either task. For the subtraction task, this might have been caused by the ceiling effect in the easy conditions: the graphs seem to indicate a clear difference between Support and No-Support. Without support there seems to be an interaction effect, whereas this effect disappears with support (performance is equal in the *hard–easy* and *hard–hard* conditions with support, see the two bottom right graphs in Fig. 2). Again, note that these data are at asymptote in the easy conditions, making it difficult to interpret these results.

In summary, presenting external support for the subtraction task reduced the effect of the problem state bottleneck for responses times of both subtraction and text-entry, but the effects did not fully disappear for either task. In the accuracy data, performance in the subtraction task seemed to reach no-bottleneck levels with support, although this was not supported by the statistical model. For text-entry there was no difference between Support and No-Support in the accuracy data.

4. Cognitive model

To account for the observed data, we incorporated the Soft Constraints Hypothesis (Gray & Fu, 2004; Gray et al., 2006) in our computational model of the problem state bottleneck (Borst, Taatgen, & Van Rijn, 2010). The model was developed in the ACT-R cognitive architecture (e.g., Anderson, 2007), and used threaded cognition theory to account for the multitasking aspects of the task (Salvucci & Taatgen, 2008, 2011). Using a cognitive architecture ensures that the components of the model have been validated earlier, which makes it meaningful to take for instance the memory, visual, and motor components of the task into account (Cooper, 2007; Newell, 1990). ACT-R has been applied successfully to explain a wide range of tasks (see http://act-r.psy.cmu.edu), and has also been mapped onto brain regions (Anderson, 2007; Borst & Anderson, 2013). We will not describe the complete model here, but refer the interested reader instead to Borst, Taatgen, & Van Rijn (2010).

For the current paper, the problem state component is our main interest. The assumption of the model is that the problem state resource can only maintain one chunk of information at a time. Thus, as long as at most one of the tasks is hard, the model can do the task without a problem – because then at most one problem state is required – but when both tasks are hard the model can only maintain one problem state, which results in interference. The model assumes that in the *hard–hard* condition, on each step of a trial the problem state resource is swapped out. That is, problem state information of the now current task is restored to the problem state resource, while problem state information of the previous task is moved to a declarative memory store. Thus, when the model switches to the other task, it first retrieves the necessary problem state information from declarative memory, restores this to the problem state resource, and only then performs the task. This takes time (a memory retrieval and 200 ms problem state restoration costs; Anderson, 2005), which results in increased response times in the *hard–hard* condition. Furthermore, incorrect problem states are sometimes retrieved from memory, resulting in lower accuracy scores in the *hard–hard* condition.

The grey bars in Fig. 2 show the fit of this model to the original task (columns one and three).[1] The model accounts well for the interaction effects in both response times and accuracy data, and also matches quite well to the absolute response times and accuracy data of the task (R^2- and RMSD-values are shown in Fig. 2). For instance, while we did not add this explicitly to the model, response times are lower in the hard text-entry condition than in the easy text-entry condition. In the easy condition the model has to read which letter it has to enter before it can search for a button and click on it, while in the hard condition the model (and subjects) already knows what word it is entering. This saves visual perception time, and thus results in lower response times in the hard text-entry condition.

We extended the model to also perform the subtraction task in the support condition. There were two basic options: either the model always uses the support indicator on the screen, which would result in equal response times in the *hard subtraction – easy text-entry* and the *hard–hard* condition, or it only uses the indicator when it cannot use its problem state, in the *hard–hard* condition. Using the problem state to remember whether a carry is in progress takes less time than having to look at the indicator on each step of a trial. Thus in total it will take less time to do the task when the support indicator is only used in the *hard–hard* condition. Given that the Soft Constraints Hypothesis assumes that humans always opt for the fastest option, we implemented this strategy in the model. Thus, the support indicator was only used in the *hard–hard* condition. This assumption led to a good model-data fit (see Fig. 2; note that all parameters were kept at the same values as for the No-Support condition). On the one hand, implementing this strategy resulted in a small interaction effect in response times in the Support condition (in the *hard–hard* condition the support indicator has to be processed, this takes more time than doing the task mentally in the *hard–easy* condition). On the other hand, it also results in a complete absence of the interaction effect in the accuracy data (as the model does not make mistakes anymore because of retrieving incorrect problem states). Thus, it seems that subjects use the externally presented support only when it helps them to do the task faster than a mental strategy would allow.

It should be noted that the model always uses its problem state resource to *process carries* in the subtraction task, also in the *hard–hard* condition *with* support. Thus, when it has to process a carry, it will use its problem state to represent the intermediate solution (e.g., when solving '6 – 4' with a carry, it will use the problem state to represent '5 – 4'). This is why the model predicts no changes to the interaction effects for the text-entry task when external support is presented: It always has to retrieve the text-entry problem state from declarative memory and restore it to the problem state resource before it can start the text-entry task. However, the data show a small decrease of the interaction effects in the support condition in the text-entry task. A simple explanation could be that subjects do not need to overwrite their text-entry problem state when using the support indicator for the subtraction task. This should lead to a complete absence of the interaction effect though, both in response times and accuracy. While we see a decrease, the interaction effect is still present. As we have no strong hypothesis about what happens, we decided against making post-hoc changes to the model to account for this.

5. Discussion

The current study investigated whether the problem state bottleneck (Borst, Taatgen, & Van Rijn, 2010) can be circumvented by providing external support. The experiment supported the hypothesis that external support can diminish the interference effects of the problem state bottleneck. However, using external support seemed to be strategic: subjects only used it when it was the fastest option. As long as a mental strategy was relatively easy (up to the *hard subtraction–easy text-entry* condition with support), subjects seemed to prefer a mental strategy to using the environment, even when support was provided. In the *hard–hard* condition, when they had to use a problem state for both tasks, they switched to environmental support when possible, resulting in faster response times with support than without support.

The model indicated that this is in accordance with the Soft Constraints Hypothesis (Gray & Fu, 2004; Gray et al., 2006), which states that humans adapt their behavior to minimize temporal costs, even if that leads to suboptimal behavior. That is what we observed in the experiment, in which it might make more sense to always use the perfect knowledge in-the-world as opposed to imperfect knowledge in-the-head. Although subjects were free to use the indicator in the Support condition – which seems to be the rational option, as it is always correct and requires less mental effort than remembering whether a carry was in progress – there was a clear difference between the *hard subtraction–easy text–entry* condition and the *hard–hard* condition, indicating that they used different strategies in these conditions. According to the Soft Constraints Hypothesis subjects tried to minimize the temporal costs of the task, and therefore prefer the imperfect knowledge in-the-head over the perfect knowledge in-the-world.

One might wonder how subjects determined the optimal temporal strategy. To discover the best strategy, the Ideal Performer Model (Gray et al., 2006) used reinforcement learning (e.g., Sutton & Barto, 1998). However, they "make no claim that the process followed by the [reinforcement learning] algorithm mimics any process followed by human cognition" (Gray et al., 2006, p. 465). Recently, Janssen and Gray (2012) investigated whether reinforcement learning could also provide a cognitively plausible explanation of the data. They concluded that reinforcement learning could be used to simulate the human data, especially if it used 'time' as the reward parameter. This is in line with other efforts to use reinforcement learning to explain human behavior (see e.g., Daw & Frank, 2009, for a special issue on reinforcement

[1] We fit the model to the data in the No-Support condition by estimating retrieval times and retrieval errors from declarative memory, and mouse- and eye movements. The model code is available from http://www.jelmerborst.nl/models/.

learning and higher-level cognition). As the subjects in our experiment received sufficient practice, it is possible that reinforcement learning can also be used to explain the strategic choices in our dataset.

Although it is not surprising that presenting external support can improve performance on a task (e.g., Hollan et al., 2000; Kirsh, 1995; Wickens, 1992), the current experiment shows that presenting information in the environment only helps in certain cases. Based on our results we can conclude two things. First, it only helps to present external support when users need their working memory for more than one chunk of information. Although it might still be used as a memory aid in other cases, there are limits on presenting information in the environment. The current research indicates that it is often not necessary to present external information. Second, even if multiple problem states are required to do a task, external support will only be used when it is faster than a mental strategy. The current, relatively simple experimental interface already showed that the costs of processing external support have an influence on behavior. This effect will be more pronounced with a real-life interface. When multiple sources of external support are present (as is often the case in real-life systems), the costs of using the support will increase, making it less likely to be used. Thus, it is important to only present external support when the costs for using it are lower than the costs of mentally doing the task would be. Using the cognitive model that we presented, it is possible to predict exactly when external support is helpful, and when not.

References

Anderson, J. R. (2005). Human symbol manipulation within an integrated cognitive architecture. *Cognitive Science*, *29*, 313–341.

Anderson, J. R. (2007). *How can the human mind occur in the physical universe?* New York: Oxford University Press.

Baayen, R. H., Davidson, D. J., & Bates, D.M. (2008). Mixed-effects modeling with crossed random effects for subjects and items. *Journal of Memory and Language*, *59*(4), 390–412.

Baayen, R. H., Piepenbrock, R., & Van Rijn, H. (1993). *The CELEX lexical, database (CD-ROM).*

Borst, J. P., & Anderson, J. R. (2013). Using model-based functional MRI to locate working memory updates and declarative memory retrievals in the fronto-parietal network. *Proceedings of the National Academy of Sciences of the United States of America*, *110*(5), 1628–1633, http://dx.doi.org/10.1073/pnas.1221572110.

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

Borst, J. P., Taatgen, N. A., & Van Rijn, H. (2010b). The problem state: A cognitive bottleneck in multitasking. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *36*(2), 363–382, http://dx.doi.org/10.1037/a0018106.

Borst, J. P., Taatgen, N. A., & Van Rijn, H. (2011). Using a symbolic process model as input for model-based fMRI analysis: Locating the neural correlates of problem state replacements. *NeuroImage*, *58*(1), 137–147, http://dx.doi.org/10.1016/j.neuroimage.2011.05.084.

Broadbent, D. E. (1958). *Perception and communication.* New York: Pergamon Press.

Carrier, L. M., Cheever, N. A., Rosen, L. D., Benitez, S., & Chang, J. (2009). Multitasking across generations: Multitasking choices and difficulty ratings in three generations of Americans. *Computers in Human Behavior*, *25*(2), 483–489, http://dx.doi.org/10.1016/j.chb.2008.10.012.

Cooper, R. (2007). The role of falsification in the development of cognitive architectures: Insights from a Lakatosian analysis. *Cognitive Science*, *31*, 509–533.

Daw, N. D., & Frank, M. J. (2009). Reinforcement learning and higher level cognition: Introduction to special issue. *Cognition*, *113*(3), 259–261, http://dx.doi.org/10.1016/j.cognition.2009.09.005.

Gillie, T., & Broadbent, D. E. (1989). What makes interruptions disruptive? A study of length, similarity, and complexity. *Psychological Research*, *50*, 243–250.

González, V., & Mark, G. (2004). Constant, constant, multi-tasking craziness: Managing multiple working spheres. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 113–120).

Gray, W. D., & Fu, W. -T. (2004). Soft constraints in interactive behavior: The case of ignoring perfect knowledge in-the-world for imperfect knowledge in-the-head. *Cognitive Science*, *28*, 359–382.

Gray, W. D., Sims, C. R., Fu, W. -T., & Schoelles, M. J. (2006). The soft constraints hypothesis: a rational analysis approach to resource allocation for interactive behavior. *Psychological Review*, *113*(3), 461–482, http://dx.doi.org/10.1037/0033-295X.113.3.461.

Hollan, J., Hutchins, E., & Kirsh, D. (2000). Distributed cognition: Toward a new foundation for human-computer interaction research. *ACM Transactions on Computer-Human Interaction*, *7*(2), 174–196, http://dx.doi.org/10.1145/353485.353487.

Janssen, C. P., & Gray, W. D. (2012). When, what, and how much to reward in reinforcement learning based models of cognition. *Cognitive Science*, *36*(2), 333–358.

Keele, S. W. (1973). *Attention and human performance.* Pacific Palisades, CA: Goodyear.

Kirsh, D. (1995). The intelligent use of space. *Artificial Intelligence*, *73*, 31–68.

McElree, B. (2001). Working memory and focal attention. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *27*(3), 817–835.

Monk, C. A., Trafton, J. G., & Boehm-Davis, D. A. (2008). The effect of interruption duration and demand on resuming suspended goals. *Journal of Experimental Psychology. Applied*, *14*(4), 299–313, http://dx.doi.org/10.1037/a0014402.

Morey, R. D. (2008). Confidence intervals from normalized data: A correction to Cousineau (2005). *Tutorial in Quantitative Methods for Psychology*, *4*(2), 61–64.

Newell, A. (1990). *Unified theories of cognition.* Cambridge, MA: Harvard University Press.

Pashler, H. (1994). Dual-task interference in simple tasks: Data and theory. *Psychological Bulletin*, *116*, 220–244.

Salvucci, D.D., & Bogunovich, P. (2010). Multitasking and monotasking: The effects of mental workload on deferred task interruptions. *Proceedings of the 28th International Conference on Human Factors in Computing Systems* (pp. 85–88).

Salvucci, D.D., & Taatgen, N. A. (2008). Threaded cognition: An integrated theory of concurrent multitasking. *Psychological Review*, *115*(1), 101–130, http://dx.doi.org/10.1037/0033-295X.115.1.101.

Salvucci, D.D., & Taatgen, N. A. (2011). *The multitasking mind.* New York: Oxford University Press.

Salvucci, D.D., Taatgen, N. A., & Borst, J. P. (2009). Toward a unified theory of the multitasking continuum: from concurrent performance to task switching, interruption, and resumption. *Human Factors in Computing Systems: CHI 2009 Conference Proceedings* (pp. 1819–1828). New York: ACM Press.

Sutton, R. S., & Barto, A. G. (1998). *Reinforcement learning.* Cambridge, MA: The MIT Press.

Wickens, C. D. (1984). Processing resources in attention, dual task performance, and work load assessment. In R. Parasuraman, J. Beatty, & R. Davies (Eds.), *Varieties of attention* (pp. 63–102). New York: Wiley.

Wickens, C. D. (1992). *Engineering psychology and human performance* (2nd ed.)New York, NY: HarperCollins.

Wickens, C. D. (2002). Multiple resources and performance prediction. *Theoretical Issues in Ergonomics Science*, *3*(2), 159–177.