Human Factors: The Journal of the Human Factors and Ergonomics Society
http://hfs.sagepub.com/

Choice in Multitasking: How Delays in the Primary Task Turn a Rational Into an Irrational Multitasker

Ioanna Katidioti and Niels A. Taatgen

Human Factors: The Journal of the Human Factors and Ergonomics Society 2014 56: 728 originally published online 26 September 2013
DOI: 10.1177/0018720813504216

Choice in Multitasking: How Delays in the Primary Task Turn a Rational Into an Irrational Multitasker

Ioanna Katidioti and **Niels A. Taatgen**, University of Groningen, Groningen, Netherlands

Objective: The objective was to establish the nature of choice in cognitive multitasking.

Background: Laboratory studies of multitasking suggest people are rational in their switch choices regarding multitasking, whereas observational studies suggest they are not. Threaded cognition theory predicts that switching is opportunistic and depends on availability of cognitive resources.

Method: A total of 21 participants answered e-mails by looking up information (similar to customer service employees) while being interrupted by chat messages. They were free to choose when to switch to the chat message. We analyzed the switching behavior and the time they needed to complete the primary mail task.

Results: When participants are faced with a delay in the e-mail task, they switch more often to the chat task at high-workload points. Choosing to switch to the secondary task instead of waiting makes them slower. It also makes them forget the information in the e-mail task half of the time, which slows them down even more.

Conclusion: When many cognitive resources are available, the probability of switching from one task to another is high. This does not necessarily lead to optimal switching behavior.

Application: Potential applications of this research include the minimization of delays in task design and the inability or discouragement of switching in high-workload moments.

Keywords: multitasking, interruption, attention, workload, delay, human-computer interaction

Address correspondence to Ioanna Katidioti, Department of Artificial Intelligence, University of Groningen, 9747 AG Groningen, Netherlands; e-mail: i.katidioti@rug.nl.

HUMAN FACTORS
Vol. 56, No. 4, June 2014, pp. 728–736
DOI: 10.1177/0018720813504216

INTRODUCTION

Multitasking has become more important in modern office work and life in general. People multitask in their cars, on the street, and while working on their computers. A key property of many multitasking situations is that people *choose* to multitask, deciding themselves to carry out tasks at the same time, or to switch from one task to another without direct external reason. The goal of this study is to investigate the process of deciding to switch from one task to another.

Observational studies have demonstrated that modern office workers switch between tasks often, on average every 3 min in a study by Gonzalez and Mark (2004), and typically require substantial amounts of time to return to a main task after they have been interrupted (Mark, Gonzalez, & Harris, 2005). Estimates have been made that 28% of a knowledge worker's day is spent on interruptions (Spira & Feintuch, 2005). Although these studies suggest multitasking leads to a loss of productivity, they cannot establish this with certainty.

The costs of multitasking have been demonstrated in several laboratory studies and are higher when mental workload is higher at the moment of interruption (Adamczyk & Bailey, 2004; Czerwinski, Cutrell, & Horvitz, 2000). The costs of interruptions can be explained by memory decay: The mental representation of an interrupted task becomes less available over time, leading to additional time requirements to either recall or reconstruct the interrupted task (Altmann & Trafton, 2007).

A limitation of almost all laboratory studies is that the interruption is forced. Gonzalez and Mark (2004), however, found that half of the interruptions observed in real office situations are self-initiated, which means that some aspect of the current or new task prompted people to switch to a different task. Even though there are some studies that show that people tend to stabilize their task before they switch (e.g., Iqbal & Horvitz, 2007), it is not yet clear what the extent of people's rationality is regarding self-interruption.

There have been some experiments that examine what happens when people have some freedom in choosing when to switch. Sellen, Kurtenbach, and Buxton (1990) showed that people tend to delay switching to another task until they complete a subtask. In a study by McFarlane (1999) participants performed better in a condition in which they were allowed to choose the moment of interruption compared to conditions where they were forced to switch immediately.

The experiment we discuss here is an extension of an experiment by Salvucci and Bogunovich (2010), where participants had the freedom to choose when to interrupt themselves. Participants had to answer e-mails as a primary task and respond to chat messages as a secondary task. They were free to switch to chat messages whenever they wanted. Salvucci and Bogunovich's results showed that people made rational choices: 94% of the switches to the chat task were made at low-workload points in the mail task. We define rational choice as in Anderson's (1990) principle of rationality, where the cognitive system tries to optimize the behavior to fit the demands of the environment, taking into account the limitations of the cognitive system.

To summarize, the two sources of information with respect to choice in multitasking contradict each other: Observation studies suggest people make poor choices in multitasking, whereas laboratory studies suggest choices are rational.

To explain choice in switching, we consider two possible accounts. The first is based on *utility*, and predicts people switch because it will lead to the best payoff in the least amount of time (we can call this global rationality). The second is based on *resource availability*, and predicts that people switch when they have many cognitive resources available (we can call this local rationality).

One prominent utility account is the soft constraints hypothesis of Gray and colleagues (Fu & Gray, 2006; Gray, Sims, Fu, & Schoelles, 2006). According to this theory, people optimize their choices in switching within a single task to maximize utility. Although not designed for multitasking, it fits the existing experimental results in multitasking very well. However, there is a conceptual problem in applying soft constraints to multitasking. The assumption of utility-based accounts is that utility is a property of the knowledge for that task, and that whenever there is a choice between two knowledge elements, the element that leads to the highest payoff for that task in the least amount of time is chosen. This means, however, that switching to another task is never attractive from a utility perspective because it only delays achieving the goal. Successful use of a utility strategy would require maintenance of a global utility in addition to utilities for the separate tasks, which may not be a tractable solution.

A resource availability account is provided by the threaded cognition theory (Salvucci & Taatgen, 2008, 2011). Threaded cognition is designed specifically to model multitasking. It assumes cognition can be subdivided into several separate cognitive resources (vision, motor, declarative memory, working memory, etc.). Particular tasks typically use only a subset of the resources and often only for certain periods, which means multiple tasks can be carried out without interference as long as they do not require the same resource at the same moment in time. The decision process is very simple: If a task needs a particular resource, and that resource is not in use by another task, it can use it.

The consequence of threaded cognition's decision process is that choice in multitasking depends on the availability of cognitive resources. If all cognitive resources are engaged with a task, the probability of switching is low, but if resources are not in use, there is a tendency to add or switch to a task that requires those unused resources. This behavior is locally rational, because it tries to use all resources as much as possible, but not necessarily rational with respect to global utility.

Threaded cognition supplies a different explanation for the rationality of choice in laboratory studies. In most multitasking experiments bad switch moments coincide with the primary task using more resources, making it less likely that a secondary task can intervene. This gives the impression people are globally rational about their multitasking. But according to threaded cognition, this rationality is based on the local optimization of resource use and therefore only local.

In the original Salvucci and Bogunovich study, global and local rationality also coincides, because high-workload periods were also the periods in which switching tasks was most detrimental to performance. The experiment we report here has added a condition to the experiment that breaks this symmetry: It introduces short periods in which the workload is very low, but switching in such periods leads to poorer performance. A utility account would predict that people still will not switch during these periods, possibly after a period of learning to learn the appropriate utilities, but a resource availability account predicts that people switch during such periods, even if this decreases their overall performance.

METHOD

Participants

Twenty-one participants (12 women), with a mean age of 23.5, were tested in all four counterbalanced conditions. There was a 5- to 10-min practice trial in the presence of the experimenter. The experiment lasted roughly 1 hr and 15 min.

Experiment

The experiment consisted of a primary task and a secondary interrupting task. The primary task was a mail task that resembled the work of a customer service employee and the secondary task was a chat message with personal questions.

In the primary task participants had to open e-mails asking them for the price of a specific product. For each e-mail the participant had to read and memorize the type, brand, and code of product (e.g., “Laptop Zanium A-63”). The names of the brands and codes were fictitious. After reading the e-mail the participant had to switch to the web browser window, obscuring the e-mail window. The initial page in this browser would show a list of product types (laptop, mp3 player, or camera), each of which would contain a link to a next page with a list of brands (e.g., “Zanium”), which were unique for each product type. Clicking on the appropriate brand name would bring the participant to a page with all the product codes for that brand (e.g., “A-63”). The pages with brand names listed three items for the mp3 players and camera categories and four for the laptop category, and the pages with product code always listed 10 items. After clicking the third link with the product code, the message “searching for price” appeared for 3 s. After this delay, the participant could read the price of the product. The participant then had to return to the e-mail window and press the “Reply” button. This resulted in the appearance of a new window (message window), where the participant could type the price and then send the message. The participant had to archive every answered message by dragging it to the “Archive” folder. That concluded the primary task after which the participant could move on to the next e-mail.

At semirandom moments, a chat message would arrive, to which the participants could switch whenever they wanted. There was on average one chat prompt per mail task. Every time a new chat message arrived, there was a notification sound and the chat window (which would be in the background but with the edge of the window visible) turned yellow. Figure 1 shows the three programs (e-mail message window, browser window, and chat window). In the real experiment, the windows always overlapped so that only one was visible at the time. Although this aspect of the interface is rather artificial, it mimics a common problem in interfaces with overlapping windows when information from one window has to be used in another window from a different application, and the screen does not afford viewing both.

This experiment is identical to the original experiment by Salvucci and Bogunovich (2010), with three exceptions. In their experiment participants had to click only two links instead of three to reach the price information (they did not have the first step of product type). This third link increases the memory load to three items. Their experiment

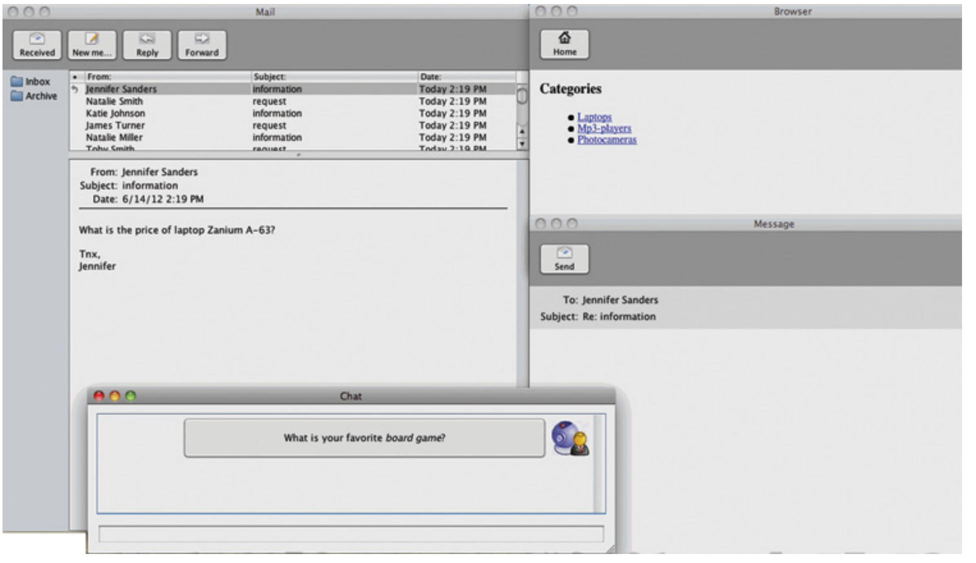

Figure 1. Screen shots of the mail, browser, chat, and message windows. During the experiment the windows were overlapping, so that the participant had to actively switch to a window to see it.

also did not include the delay before the price appeared. We added this delay to create a clear low-workload moment in the middle of the task, a moment where working memory does not contain either the product name or its price. The third modification is that we added another delay in the e-mail program: Whenever the participant would switch to the e-mail window, it took 3 s to show the message in that window.

In addition to the basic task, we added two experimental manipulations to the design: the presence or absence of a 3-s delay after clicking the first two browser links (delay/no delay conditions) and the difficulty of the questions in the chat task (difficult/easy conditions). In the delay condition, there would not only be a delay after clicking the third link in the browser, but after clicking any of the links. After the participant clicked the first (product type) and second (brand) link in the browser, a "loading" page would be shown for 3 s before proceeding to the requested page that the link referred to.

The key characteristic of this design is that switching during the first or second link requires maintenance of the product information in working memory, while switching during the third link does not. If a chat message arrives during Link 1 or 2, it is therefore better to postpone answering it until Link 3 has been clicked. The delay in the e-mail program was added to create an extra penalty for those who forgot the product information and had to return to the e-mail and read it again. If participants switched during a delay on Link 1 or Link 2 (trying to take advantage of the delay time), but then forgot the product information and had to turn back to the e-mail and read it again, they would have to wait 3 more s and eventually lose more time. We hoped that this would make participants more responsible when deciding to switch while they had to retain information on their working memory (high-workload moments).

Figure 2 shows the sequence of steps in the mail task with an indication whether or not information needed to be retained at that point. All events involve a mouse click, except "Compose Type," where a message has to be typed in the message window, and "Link Request 1-3," where the participant has to wait for the appearance of a webpage. There are moments of high workload (gray boxes), when the participant's working memory contained either the product description or the product type, and moments of low workload (white boxes). The chat messages appearances were equally distributed in high and low workload moments.

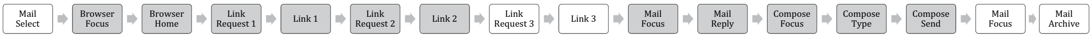

Figure 2. The sequence of events of answering an e-mail. The white blocks represent moments of low workload and the gray ones moments of high workload, as the participant had to remember either the model or the price of the product.

The second manipulation we introduced concerned the difficulty of the chat messages. In the easy chat condition, participants were asked questions about movies ("Have you seen the movie . . . ?"), which required recognition and a yes/no answer. In the difficult chat condition participants were asked for a favorite book, artist, CD, and so on ("What is your favorite . . . ?"), which required recall and are open-ended questions. One third of the questions on each condition were follow-up questions related to the previous question or answer, giving the chat task a more realistic form and making participants pay more attention to the questions. In the easy condition the follow-ups included asking "Did you like it?" if the respondent had answered "yes" to the previous question or "Do you want to see it?" if the respondent had answered "no." In the difficult condition the follow-up question was "What is your least favorite?" so the participant had to remember what the first question asked. Each participant had to complete four blocks: delay/difficult, delay/easy, no delay/difficult, no delay/easy. The goal of the difficulty manipulation was to see whether difficult questions would lead to additional disruptions in the primary task, possibly encouraging participants to avoid switching during high workload.

The blocks appeared in counterbalanced order. Each block was completed after the participant had answered 24 chat questions (8 of them were follow-up questions), so the duration of the experiment depended on the participant's choices. However, all the participants completed the four blocks in roughly 1 hr. Participants were instructed to give equal priority to both tasks.

RESULTS

The difficulty of the chat task variation (difficult and easy conditions) did not produce any significant results in either time or switching behavior of the participants, so we do collapse the data over this condition and only analyze the effect of browser delay (delay and no delay conditions).

We analyzed participants' switching behavior by counting the number of switches to the chat task after each mail task event. We included in the analysis only the chat prompts that appeared at a high-workload point, because those were the ones that created interference to the participants. The majority of chat prompts that appeared on low-workload moments were answered immediately or in the next step (if it was also a low-workload point). The proportion of switches made by all participants after each event are shown in Figure 3.

During the browser delay that occurs in the delay conditions (events "Link Request 1" and "Link Request 2"), participants were tempted to switch to the chat task, even though it is a moment of high workload. In contrast, there was only one change on those two events in the no delay conditions. The percentage of switches on high workload points was higher in the delay conditions (27.2 %, $SE = 5\%$) than in the no delay conditions (9.5 %, $SE = 3.2\%$). A paired t test on arcsin transformed proportions shows this difference is significant, $t(20) = 4.43$, $p < .001$, $d = 1.12$.

Closer inspection of the data revealed some individual differences: All but 4 respondents made at least one switch to the chat task during a high workload pause, but some more than others. However, there was no evidence of any change in switching behavior over the course of the experiment, so no evidence for learning.

Participants switched at high-workload points in the delay conditions, but did that affect their performance? To measure their performance we analyzed the average mail task time, not including the browser delays in the delay conditions and the chat task time in both conditions. In the delay condition participants spent on average 24.2 s on each mail ($SE = 0.87$), whereas in the no delay condition they needed only 21.9 s

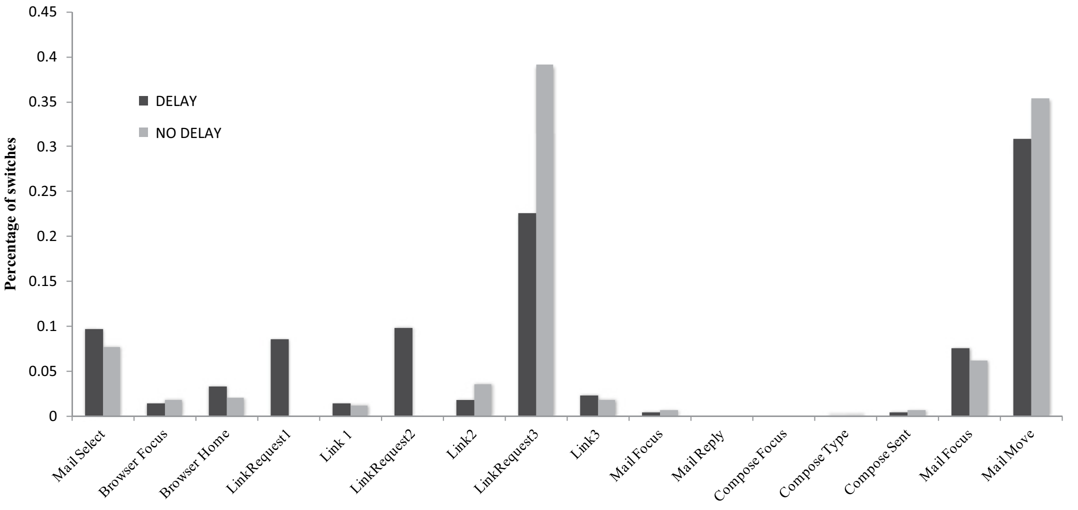

Figure 3. Percentage of switches to the chat task after a mail task event for the delay and no delay conditions.

(*SE* = 0.75). A paired *t* test shows this difference is significant, $t(20) = 4.45$, $p < .001$, $d = 0.69$. Participants answered on average 25.1 e-mails in the delay conditions and 25.2 in the no delay conditions, which is not significantly different, and they switched to the chat task approximately once during every e-mail.

The participants' decision to switch at high-workload points during the delay conditions made them significantly slower in the primary task (the mail task). However, since they were going to answer the chat message at some point, choosing to do so during a forced pause in the primary task and not later could mean that they used the browser delay time productively. Since an amount of time is required to answer the chat message, why not do it during the delay and gain 3 s? To determine whether switching during moments of high workload had indeed a negative effect on the participants' performance, we compared the time they lost due to switching (which is the time difference between the delay and no delay conditions) with the gain they had from answering chat messages during the delays.

The time that participants lost because of switching on high-workload moments is the mail task time of the delay conditions (24.2 s) minus the mail task time on the no delay conditions (21.9 s), which is 2.3 s. Participants made a total of 178 switches to the chat task during high-workload moments. We included only the switches that occurred during the first high-workload moments (Link Request 1 and Link Request 2), because there was the delay variation. The time they gained is 3 s per switch, which is, given that a total of 1055 e-mails were answered in the delay conditions, 0.51 s per e-mail (which is demonstrated by the line in the delay bar in Figure 4). These results show that participants lost more time (2.3 s) than they gained (0.51 s). A paired *t* test shows the difference is still significant, $t(20) = 3.49$, $p = .002$, $d = 0.54$, confirming that switching on high-workload points has a negative effect on performance.

After 97 out of the 178 switches (55%) that occurred during high-workload moments, participants had to return back to the mail window and read the product model again, whereas in the other 96 high-workload switches, they were able to recall the information and didn't have to return. The analysis done by using linear mixed-effects models showed that participants were significantly slower ($\beta = 3.11$, $t = 3.59$, $p = .0002$) in the mail task for the trials in which they forgot the information and had to go back ("Return" column in Figure 5) than the trials in which they memorized the information ("No Return" column in Figure 5). The delay of the mail window was removed for the second time they returned to read the e-mail. Still, even if they didn't forget the information, they were significantly slower from the no delay conditions ($\beta = 11.01$, $t = 8.84$, $p = .0001$).

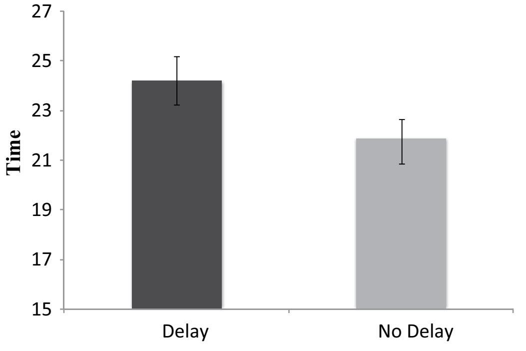

Figure 4. Average time per e-mail in the mail task for the "delay" and "no delay" conditions. The line on the delay bar shows the time gained due to switching during the browser delays.

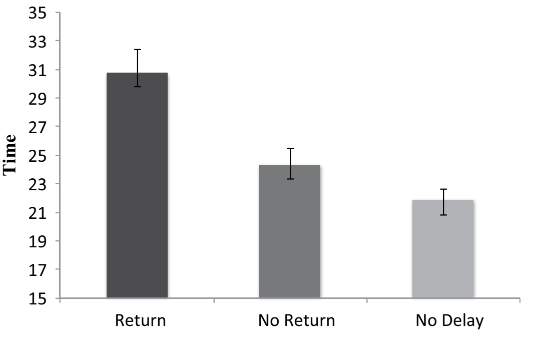

Figure 5. Average time per e-mail in the mail task for the situations where the participants returned to the mail window to read the e-mail information again after a bad switch ("return"), for the situations where they didn't return after a bad switch ("no return"), and for the "no delay" situations, where there were not bad switches.

DISCUSSION

Do people switch tasks to optimize utility, or do they change tasks because they have available cognitive resources? The results of the experiment favor the latter explanation. If 3-s pauses are introduced during periods of high memory load, people tend to switch tasks even though this leads to overall decreases in performance. The extra time cost is partly incurred because participants forgot information and had to reread it. But even in cases in which participants did manage to remember the information, they were still worse off than when they delayed switching until memory load was zero.

The results are not consistent with the theory of utility maximization. A utility account would also predict that switch behavior improves as people discover utility values. However, no learning is found in the data. The results do agree with threaded cognition's "greedy" theory (Salvucci & Taatgen, 2011), which states that people will switch to a task that is waiting as soon as the resources for it are available. Salvucci and Taatgen (2011), following Altmann and Trafton (2007) and Borst, Taatgen, and Van Rijn (2010), explain why switching during working-memory load decreases performance: When people return to the primary task after a switch, they have to restore their working memory. Therefore, when participants choose to switch tasks when their working memory contains information important for the task (as they did in the delay condition) they need more time to restore that, compared to switching when their working memory contains nothing vital (as they did in the no delay condition).

One aspect that the resource availability account cannot fully explain is individual differences, in particular individuals who never switch during the pauses and individuals who do this only a few times. According to threaded cognition, when people are doing the experiment they create a goal for the mail task. But as soon as the chat window indicates there is a message, they create a parallel chat goal. The chat goal will compete for resources with the mail goal, but as long as the mail goal occupies most resources, the chat goal will not be able to interrupt. That is why almost no switches occur during the high-load period in the control condition. However, when there is a pause in the browser, all resources except working memory are available, and therefore people switch to the chat goal. There are some possible explanations for why people do not switch. If they are actively rehearsing information during the pause their resources may be sufficiently occupied so that they do not switch. Alternatively, nonswitching may be a strategic choice that people have picked up in other multitasking situations. In that case there is a tendency to switch goals, which is then overridden by an explicit strategy (like dieters consciously suppressing the tendency to eat available food). Whichever explanation holds, it is not a strategy that develops during the experiment, because otherwise we would see effects of learning. This study by itself is not enough to support any of these explanations. However, the results are in line with observational studies in which unproductive interruptions are common, and where people try, with various levels of success, to control themselves.

PRACTICAL IMPLICATIONS

The results of our experiment suggest that a delay in the primary task is a strong trigger for switching to a secondary task. Given that the participants would switch to the secondary task at some point, doing so during a delay seems like a good decision. However, the results show that switching tasks when working memory has to sustain information is detrimental for the performance on the primary task. Participants either forget the information and have to reread it, or remember the information but still need more time to finish the primary task than when they switched at low-workload points.

This research underlines the importance of cognitive task analysis in design. A recommendation is to avoid delays during high-workload moments in task execution. If delays cannot be avoided by hardware or software means, it is better to insert delays at low-workload moments. A second recommendation is to discourage switching. That can be accomplished by making potentially distracting tasks less visible, making it less likely that the user would want to pursue them. Another option for discouraging switching is to keep the user engaged during the delay, preferably with something that helps retaining the information in working memory. Finally, given that not all participants made bad switches in the delay condition, it appears people can be trained on proper switching behavior, and it might be worthwhile to investigate this.

ACKNOWLEDGMENTS

This research was funded by ERC-StG Grant 283597 awarded to Niels Taatgen. We thank Dario Salvucci for making the code for the experiment available to us.

KEY POINTS

- When people are given a free choice of when to switch from a primary task to a secondary task, they tend to switch at moments when cognitive resources are available for the secondary task.
- The availability of cognitive resources typically coincides with a low working-memory load, which leads to switching at appropriate moments.
- If there are short (3 s) delays in the primary task, this prompts people to switch tasks during the delay, even if this is at a moment of high working-memory load.
- When people switch during a moment of high working-memory load, they either forget their primary task context (about 50% in our experiment) or need time to recall it.
- Regardless of whether people forgot the context of the primary task, they were slower than when they would have waited.
- This research shows that people's multitasking decisions are locally rational, but can still be suboptimal on a global level.

REFERENCES

Adamczyk, P. D., & Bailey, B. P. (2004). If not now, when? The effects of interruptions at different moments within task execution. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 271–278). New York, NY: Association for Computing Machinery.

Altmann, E. M., & Trafton, J. G. (2007). Timecourse of recovery from task interruption: Data and a model. *Psychonomic Bulletin and Review*, *14*, 1079–1084.

Anderson, J. R. (1990). *The adaptive character of thought*. Hillsdale, NJ: Lawrence Erlbaum.

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

Czerwinski, M., Cutrell, E., & Horvitz, E. (2000). Instant messaging and interruption: Influence of task type on performance. In C. Paris, N. Ozkan, S. Howard, & S. Lu (Eds.), *OZCHI 2000 conference proceedings* (pp. 356–361). New York, NY: Association for Computing Machinery.

Fu, W.-T., & Gray, W. D. (2006). Suboptimal tradeoffs in information seeking. *Cognitive Psychology*, *52*, 195–242.

Gonzalez, V. M., & Mark, G. (2004). "Constant, constant, multi-tasking craziness": Managing multiple working spheres. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 113–120). New York, NY: Association for Computing Machinery.

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*, 461–482.

Iqbal, S. T., & Horvitz, E. (2007). Disruption and recovery of computing tasks: Field study, analysis and directions. In

Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (pp. 677–686). New York, NY: Association for Computing Machinery.

Mark, G., Gonzalez, V. M., & Harris, J. (2005). No task left behind? Examining the nature of fragmented work. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 321–330). New York, NY: Association for Computing Machinery.

McFarlane, D. (1999). Coordinating the interruption of people in human-computer interaction. In *Human-computer interaction—INTERACT '99* (pp. 295–303). Amsterdam, Netherlands: IOS Press.

Salvucci, D. D., & Bogunovich, P. (2010). Multitasking and monotasking: The effects of mental workload on deferred task interruptions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 85–88). New York, NY: Association for Computing Machinery.

Salvucci, D. D., & Taatgen, N. A. (2008). Threaded cognition: An integrated theory of concurrent multitasking. *Psychological Review*, *115*, 101–130.

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

Sellen, A. J., Kurtenbach, G. P., & Buxton, W. A. S. (1990). The role of visual and kinesthetic feedback in the prevention of mode errors. In *Proceedings of the Third IFIP Conference on Human-Computer Interaction (INTERACT)* (pp. 667–673). Cambridge, UK: Springer.

Spira, J. B., & Feintuch, J. B. (2005). *The cost of not paying attention: How interruptions impact knowledge worker productivity*. New York, NY: BASEX.

Ioanna Katidioti is a PhD student at the University of Groningen. She earned her MSc in basic and applied cognitive science from the University of Athens in 2012.

Niels A. Taatgen is full professor of cognitive modeling at the Department of Artificial Intelligence of the University of Groningen. He earned his PhD in psychology from the University of Groningen in 1999.

Date received: September 10, 2012
Date accepted: August 15, 2013