On the consideration of learning styles in assessing learners' cognitive load during learning

¹Abdul-Rahman, S-S, ²du Boulay, B.

¹Faculty of Computer Science and Information Technology, University of Malaya ¹siti_soraya@um.edu.my ²Department of Informatics, University of Sussex ²B.Du-Boulay@sussex.ac.uk

Abstract

Learning style is a factor in determining how much effort learners are willing to expend in studying worked-examples. This paper investigates variations of cognitive load during learning, taking into account the learner's learning style. A web-based worked-example system for learning programming was used for the purpose of investigations.

Keywords: Cognitive load, learning styles, worked-examples, programming

1. Introduction

The design and use of worked-examples have been widely studied within cognitive load research. However, few of these studies provide conclusive findings for their effectiveness either in terms of the learning *process* or its *outcomes* or indeed in terms of cognitive load effects [5].

Cognitive load research has generally studied the effects of cognitive load between participants in different treatment groups [3]. One problem associated with such an approach is how to achieve equal variances (i.e. individual differences) among the treatment groups. Thus, [3]'s recommendation is "...to include these individual characteristics as control variables in experimental set-ups so that differences between groups cannot be attributed to differences in these relevant student characteristics" (pp. 123). In line with this, [6] argued that learning style plays a major role in determining how much "...effort learners are willing to expend in understanding worked-examples, with active learners tending to be more impatient than reflective learners" (p. 5). Active learners can be forced into a more reflective style of learning, and as a result, they may experience cognitive *over(load)*, to use [3]'s phrase. Note that, central to cognitive load theory (CLT) is a focus on the design of instructional materials that align appropriately with the learner's limited working memory capacity [7].

Several empirical findings within the programming education literature have revealed that reflective learners outperform active learners in introductory computer science courses and/or in programming performance (e.g. [1][2][8]). Among others, [9] has studied the effects of two instructional strategies, namely program completion and program generation on the reflectivity-impulsivity dimension of cognitive style. A compensatory model was used to test the hypothesis that the program completion strategy could compensate for the (supposed) negative effects of impulsivity among active learners. However, the results revealed no support for this model. On the other hand, in a subsequent long-duration experiment, the results revealed a relationship between the instructional strategies and reflectivity-impulsivity, hence, providing support for the preference model.

2. Goal of the work

This paper describes an experiment that investigated any consequential effects of the learner's cognitive load in learning programming via worked-examples, taking learning style into account. More specifically, the study was conducted to test a tentative hypothesis and the findings of the first author's thesis work, i.e. learning style might interact with learner's cognitive load [6]. For the purpose of the investigation, a web-based worked-example system for learning programming (LECSES) was used and this system offered learners a worked example strategy called the Completion strategy [10]. This strategy helps learners gradually build up a schema by getting them to complete a partial solution to a worked-example [10].

3. Method

An experiment was conducted with 33 university students who enrolled in an introductory programming course at a university in Malaysia. The participants were grouped according to their learning style, namely an active and a reflective group. The participants worked in a web-based learning environment, LECSES [10]. The experiment took place in two separate phases, that is, a learning phase and a transfer phase.

In the learning phase, the participants were asked to study a pair of isomorphic worked-example problems. In particular, they had to complete a partial program solution to each worked-example problem. Each problem consisted of two exercises and a total of 30 minutes (15 minutes per exercise) was allocated for the learning time. Thus, the overall learning time was 60 minutes. After working with each problem, participants were asked to give an estimate of the difficulty of the learning method (on a scale ranging from 1 "Very easy" to 5 "Very difficult", as adapted from [4]) and of the degree to which they had learned new Java concepts from the materials (on a scale ranging from 1 "Not at all" to 5 "Verv much").

In the transfer phase, a program development and coding test was administered to measure the participants' transfer of their knowledge. The participants were asked to solve two programming problems, distinguishing between near- and far-transfer. The near-transfer problem was similar in underlying structure to the worked-examples studied in the learning phase but had a different surface story. The far-transfer problem had a different structure from the worked-examples encountered in the learning phase and required different problem solving techniques. After solving each transfer problem, participants were asked to give an estimate of the their mental effort in solving the problem (on a scale ranging from 1 "very low mental effort" to 5 "very high mental effort").

In the study, the two groups were compared in terms of their perceived effort and difficulty during learning, estimated mental effort on post-test, and their performance on the problems in the transfer phase.

4. Analysis of results

4.1 Learning phase

The Mann Whitney U test revealed that the effort scores of the active learners did not differ significantly from the reflective learners (see Table 1), U = 71.00, Z = -1.31, ns, r = -.24. With regard to the difficulty scores, the result of the U-test was likewise non-significant, U = 104.50, Z = .00, ns, r = -.00.

Table 1: Descriptive statistics for the dependent variables				
	Active (n=??)		Reflective (n=??)	
	Mean	Std. Deviation	Mean	Std. Deviation
Pre-test	8.43	2.06	9.54	3.15
Effort	6.30	1.81	5.10	2.23
Difficulty	7.74	1.33	7.73	1.62
Mental Effort	7.90	1.62	6.64	3.23
Transfer performance	4.38	3.10	5.54	2.99

Note: Pre-test score is taken as an average of two tests. Transfer performance (0 - 14), i.e. 7 marks for each transfer problem.

4.2 Transfer phase

A Mann Whitney U test revealed no significant difference between the active learners and reflective learners on their mental effort scores, U = 90.00, Z = -.84, ns, r = -.15. With regard to their transfer performance, the U-test also found no significant difference, U = 99.00, Z = -1.01, ns, r = -.18.

5. General discussion and conclusion

The absence of significant effects of learner's cognitive load during the transfer phase is disappointing in that it was not possible to support the tentative hypothesis. Then again, whereas [9]'s findings indicated little support for the idea that the program completion could be compensated for the (supposed) negative effects of impulsivity, the present findings appeared to suggest otherwise. In particular, the Completion strategy used in this study seemed equally suited to both the active and the reflective learners who seemed to benefit equally from it.

Even so, it is worth noting that a similar earlier study [6] found trends in its data, statistically nonsignificant but with a small to medium effect size, suggesting that in the learning phase reflective learners tended to report higher levels of difficulty in studying worked-examples using the Completion strategy than active learners. In addition, the trends also suggersted that reflective learners tended to report higher levels of mental effort in solving the transfer tests than active learners. For an extensive review of this evidence, see [6].

One caveat remains, that the sample size of this study was too small, with an unequal number of active and reflective learners, and so merits further research and should be conducted with a larger sample size to corroborate this claim.

As a final point, it is argued that investigating the relationship between learning styles and teaching method, and so the consequential effects of cognitive load on active and reflective learners remains a promising area for further investigation and should be taken into account when designing CLT experiments.

6. References

- [1] Allert, J., Learning style and factors contributing to success in an introductory computer science course, *Proceedings of the IEEE International Conference on Advanced Learning Technologies* (pp. 385-389), IEEE, 2004.
- [2] Chamillard, A.T., & Karolick, D., Using learning style data in an introductory computer science course, *Proceedings of the thirtieth SIGCSE technical symposium on Computer science education* (pp. 291-295), New York, USA: ACM, 1999.
- [3] de Jong, T., Cognitive load theory, educational research, and instructional design: some food for thought, *Instructional Science*, *38*, 105-134, 2010.
- [4] Kalyuga, S., Chandler, P., & Sweller, J., Levels of expertise and instructional design, *Human Factors*, 40, 1-17, 1998.
- [5] Moreno, R., When worked examples don't work: Is cognitive load theory at an impasse? *Learning and Instruction*, *16*, 170-181, 2006.
- [6] Siti-Soraya, A-R., *Learning programming via worked-examples: the effects of cognitive load and learning styles*, Doctoral thesis, University of Sussex, 2012.
- [7] Sweller, J., van Merriënboer, J.J.G., & Paas, F., Cognitive architecture and instructional design, *Educational Psychology Review*, 10, 251-296, 1998
- [8] Thomas, L., Ratcliffe, M., Woodbury, J., & Jarman, E., Learning styles and performance in the introductory programming sequence, *SIGCSE Bulletin*, *34*, 33-37, 2002.
- [9] van Merriënboer, J.J.G., Instructional strategies for teaching computer programming: Interactions with the cognitive style reflection-impulsivity, *Journal of Research on Computing in Education*, 23, 45-53, 1990.
- [10] van Merriënboer, J.J.G., & de Croock, M.B.M., Strategies for computer-based programming instruction: Program completion vs. program generation. *Journal of Educational Computing Research*, *8*, 365-394, 1992.