A Computational Model of Retrospective Time Estimation

Jeronimo Dzaack (jdz@zmms.tu-berlin.de)
User Modeling in Dynamic Systems, Technische Universität Berlin
Jebensstraße 1, J2-21, 10623 Berlin, Germany

Sandra Trösterer (str@zmms.tu-berlin.de)
User Modeling in Dynamic Systems, Technische Universität Berlin
Jebensstraße 1, J2-2, 10623 Berlin, Germany

Nele Pape (npa@zmms.tu-berlin.de)
User Modeling in Dynamic Systems, Technische Universität Berlin
Jebensstraße 1, J2-2, 10623 Berlin, Germany

Leon Urbas (lur@zmms.tu-berlin.de)
User Modeling in Dynamic Systems, Technische Universität Berlin
Jebensstraße 1, J2-2, 10623 Berlin, Germany

Abstract

Retrospective time estimation is an important aspect in dynamic systems and needs to be integrated in cognitive architectures. In this article a short overview of theoretical accounts of retrospective time estimation is given and assumptions based on an experiment conducted in our research group are presented. Regarding both aspects we introduce a retrospective timer-module for ACT-R 6.0 and the corresponding estimation algorithm. The successful validation of the module is shown and further implications are discussed.

Introduction

Estimation of time-duration plays an important role whenever it is necessary to be aware of a sequential occurrence of events. Especially in dynamic human machine systems the estimation of time-duration is an essential requirement for system control (Schulze-Kissing et al., 2004). In some situations processing temporal information is the only means to provide information on critical system incidents or abnormal system performance. Therefore time-duration estimation should be considered as an important aspect in designing human machine interaction (HMI).

Psychological research distinguishes between prospective and retrospective duration estimation (James, 1890, c.f. Hicks et al. 1976). In prospective settings, the subject knows in advance that duration is important, whereas the subject is not informed that duration estimation is of interest in retrospective settings.

Despite the fact that both duration estimation methods (and possibly a mixture of them) play an important role in HMI, relatively few studies have used a retrospective paradigm.

A concise theory and a computational implementation of retrospective duration estimation would be an important component of cognitive architectures like ACT-R (Anderson & Lebiere, 1998), Soar (Laird & Rosenbloom, 1996) and 3Caps (Just & Carpenter, 1992) for HMI engineering. Therefore, we derived an approach of retrospective time-duration estimation from literature data and data gained from an own experiment. We developed an algorithm that takes into account theoretical assumptions of retrospective time-duration estimation and own considerations based on empirical data. We further implemented the algorithm as a timer-module in ACT-R 6.0 to provide a tool for modeling retrospective time estimation and validated it against the experimental data.

Retrospective Time Estimation

"In the retrospective paradigm, a person becomes aware of the need to judge a duration only after it has ended." (Zakay & Block, 2004, p. 320).

Different approaches try to explain how people estimate time in retrospect. Agreement exists that time is underestimated in relation to the experienced time-interval. Most authors claim that estimation time depends on memory encoding, storage or retrieval processes. Block & Reed (1978) proposed the contextual-change hypothesis, which asserts that remembered duration is mediated by the remembered amount of changes in cognitive context during an interval.

Among others, McClain (1983) showed that the duration of retrospective judgments increases with the amount of items encoded. In McClain's experiment subjects had to process 15, 30 or 45 words within 120 seconds and to allocate them to one of two given groups. She assumes a direct function to explain this dependency of responses and estimated time.

A meta-analysis by Block & Zakay (1997) shows that retrospective time estimation leads to larger variance in the estimations compared to the prospective time estimations, especially for long durations.

In our own experiment (Pape et al. 2005) we found that variance of reproduced duration estimation increases with growing waiting time (nothing is done by the subjects to reach the current goal) in a static time interval. We assume this waiting time as *idle time*: that means no task is executed and no perceptual changes occur. Idle time seems to have an effect on the dependency of responses and duration estimation.

A GOMS-analysis (Card, Moran & Newell, 1980) of McClain's experimental task with the software Travis (Hamacher, Kraiss & Marrenbach, 2002) reveals that processing a word takes about one second (attend, read, decide between two categories, respond). So participants experience different amounts of waiting time depending on the number of words in the 120 seconds interval. In McClain's experiment no variance-index is given, but in connection with our experimental data we assume a growing variance of time estimations with growing idle time in a static interval.

Computational Model

In the previous section we showed three main aspects that influence retrospective time estimation. Shortly: (1) the existence of contextual changes, (2) a direct function between number of responses and retrospective duration judgment and (3) an enhancement of variance of estimations due to the idle time portion of a given time-interval.

We derived the following equation from the first and the second assumption:

Duration Estimation = C x Responses x Total Task Time

That means that the duration estimation depends on the total task time, the number of responses and a parameter C that has to be calculated from empirical data and moderates the dependency. Simplifying, we assume that the number of responses equals the number of contextual changes in the given total task time.

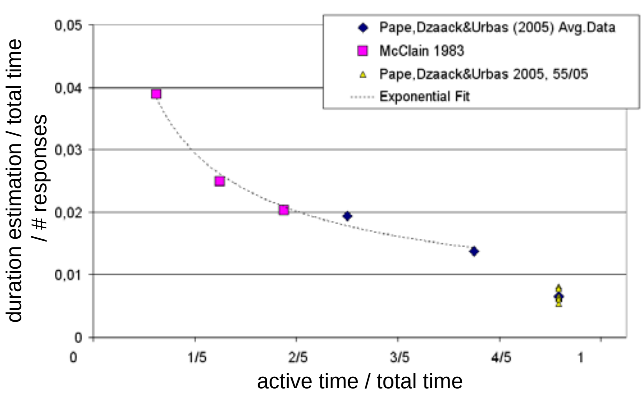

Figure 1: The power function

Based on this equation we took a further look on the data of McClain (1983) and data derived from the experiment conducted by our research group. In our experiment we had three conditions with 60 seconds total task time and different amounts of idle time respectively (5, 15, 30 sec., see Pape et al., 2005). In each condition 10 subjects participated and had to reproduce the time interval. We found that for large idle-times (15, 30 sec.) no linear function, as assumed by McClain (1983), exists that describes the data satisfactorily (r^2=.27, r^2=.12 vs. r^2=.71 for 5 sec. idle-time). Therefore we conclude that the equation is not adequate to describe our experimental data. To do justice to these cases we changed the static parameter C to an equation that calculates a dependency on the idle time portion: If idle time decreases, the ratio of active time to total time increases and the influence of the idle time is lowered. Contrasting this ratio with the duration judgment ratio (i.e., duration estimation / total time) normalized by the number of responses, a power function can be derived that explains McClain's and our data (see Figure 1). The term can be interpreted as the probability of a positive correction of the time-duration estimation.

For small portions of idle time no connection to the power function can be found, so we assumed the existence of a threshold to describe all data. Regarding the experimental data it can be assumed that this threshold is about 10% idle time of total task time.

Regarding all these aspects we developed the following equation to reproduce retrospective time estimation judgments:

$$DE = \begin{cases} (A + B \times (AT/TT)^{-0.5}) \quad \times R \times TT & ; AT/TT < 0{,}9 \\ \\ C \quad \times R \times TT & else \end{cases}$$

Figure 2: Duration Estimation Algorithm
(DE: duration estimation, R: # Responses, AT: active time, TT: total time, A, B, C: parameters fitted to empiric evidence)

That means, if the ratio of active time to task time is greater than 10%, idle time has an influence on the duration estimation in our computational model of retrospective time estimation. The parameters A, B and C are derived from our experiment conducted at the Technische Universität Berlin and McClain's data. The constant A represents the influence of the passed time moderated by the responses just the same way as C does for the time-duration estimation without idle time. The constant B represents the influence of the corrective processes that are triggered by the experience of idle time intervals.

Concept and Aspects

A time-estimation component in a cognitive architecture has to be independent from the task and the model itself to allow a self-contained application of the component. All functions have to be encapsulated and functions to access this component have to be provided.

The retrospective time estimation approach can be compared with an abstracted episodic memory store. New episodic reference-points can be set to split up the passed time. In the retrospective approach this is done by the explicit setting of distinctive reference-points according to reality (i.e., distinctive actions will be held in memory as landmarks and help to navigate through the passed time). The time between these reference-points is estimated by the duration estimation algorithm (see figure 2). If the model should estimate the time between two points a reference-point has to be set. From this reference-point the retrospective time is estimated. If the endpoint is reached, a new reference-point has to be set by the model (the modeler). In this case all processed information for the current reference-point is stored in a memory element (creation time of reference-point, estimation, current time and responses during the interval) and a new reference-point is created (see figure 3). From this point a new estimation is made either to the end of the simulation or until a new reference-point is set. All estimations are accessible during the simulation through the memory elements or the module itself.

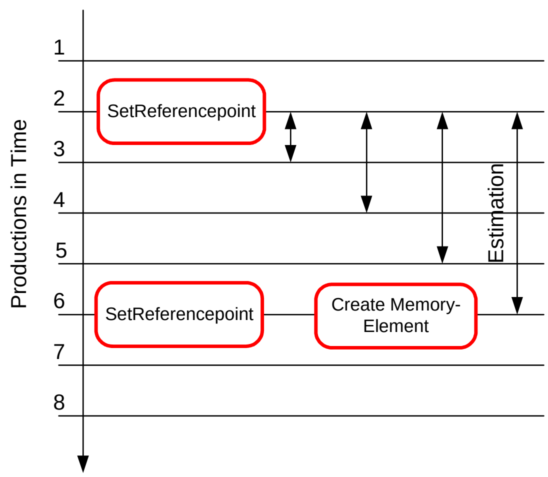

Figure 3: Retrospective time estimation

Because of the normalization of the parameters A, B and C of the algorithm, these can be handled as general parameters that are independent from contextual changes. In this approach all fired productions that are needed to do the task (active productions) are counted and represent the factor R in the algorithm. To distinguish between active and idle productions (e.g. waiting), idle productions have to be marked as idle by the modeler.

To measure the different times (active time and idle time) the time between two productions is allocated to the state of the first production (idle or active) and is added to the particular portion of time. The total time is the time between the instantiation of the reference-point and the current system time.

Implementation

We developed a retrospective timer-module for ACT-R 6.0 (Anderson & Lebiere, 1998; Bothell, 2005). ACT-R (atomic components of thought) is a cognitive architecture based on a production system structure. It features assumptions of human cognition derived from psychological concepts. Based on the theory ACT-R provides components (modules, buffers and chunks) that allow the simulation of human cognition. These components are managed in a special layer with own constructs (models, events etc.).

Modules encapsulate operations and methods belonging to a specific aspect of cognition and are independent from the modeler’s perspective. To access these functions a buffer for each module is needed. The buffer provides methods to access and manipulate the stored data. Therefore the buffer uses the provided methods of the module. ACT-R represents the knowledge in terms of chunks, either in a buffer or the integrated declarative memory. To run a simulation a set of production rules is needed. A production is activated in respect of the current state of the system (i.e., the state of the buffers) and through its activation the state of the system changes.

The algorithm for retrospective time estimation is implemented as a module in ACT-R and encapsulates all implementation details. The timer-module code has to be placed in the framework (folder: *modules*). Compiling the code provides a timer-buffer that allows the modeler a uniform access to the timer-module. Table 1 summarizes the four statements that are necessary to use the timer-buffer in ACT-R.

Table 1: Commands for the use of the timer-buffer in ACT-R

RHS:
`+timer>`
`    isa       timer-reference`
`    mode      retro`
`    id        =id`
	Set a timer-reference

LHS:
`=timer>`
`    isa       timer-duration`
`    id        =id`
`    duration  =duration`
	Access a retrospective duration estimation

RHS:
`-timer>`
`    isa       timer-reference`
`    id        =id`
	Delete a timer-reference

RHS:
`?timer>`
`    isa       timer-reference`
	Question the state of the timer

The timer-module works as follows: If a reference-point is set, every time a production rule fires the estimation is calculated by the algorithm. Therefore all factors of the equation have to be updated by the module. If a production is not marked as idle, the amount of active productions (R) is increased by one. Then the amount of time between the previous and the current production is added to the active or the idle portion of time (active time or idle time) in respect to the state of the previous production. The total time of the actual timer-reference is updated. Subsequently the state of the current production (active or idle) is stored. After the instantiation of a reference-point (the first production in a time interval) no time is measured, because no clue is available whether this time is active or idle. In the following production the missed amount of time is added to the appertain interval. After updating all required data the estimation algorithm is performed. Therefore the active time (AT) to total time (TT) ratio is calculated and the corresponding case is selected, calculated and finally the information in the buffer is updated.

Model Data vs. Empirical Data

To validate the implementation details, we implemented a model of the interaction of humans with the D2-Drive test (Urbas et al., 2005) which we adapted for the experimental design of the study described before. The D2-Drive refers to the D2 test of attention by Brickenkamp (2001) that investigates individual differences on attention and concentration. The aim is to identify a pattern as correct according to Brickenkamp´s specification (a *d* with two strokes). As in real-life-scenarios of human-machine-interaction, e.g. when managing a navigation system or switching the radio, the used test acquires visual attention and can be seen as a model for a driver infotainment system. The model runs the D2-Drive for a given time (55, 45, 30 seconds) and then changes to an idle mode that emulates waiting to fulfil the 60 seconds used in the experimental design. We added the timer-module to the framework of ACT-R and run the model with the integrated timer.

As intended, the ACT-R model with and without the integrated timer-module does not show any differences in performing the task. This was measured by the given responses of identifying the pattern (i.e., while doing the active task).

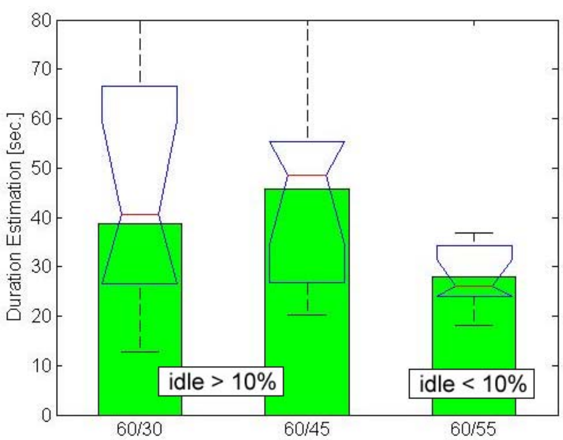

Figure 4: ACT-R model vs. empirical data of 3 Groups of people

Running the ACT-R model with the timer-module shows the retrospective estimation of time-duration as anticipated: the duration estimated by the ACT-R model interacts with the amount of idle time and the given responses as observed in the experiment. In figure 4 the experimental data is represented by boxplots and the module's estimates by bars. It can be seen that our timer-module reproduces the tendency of our experimental data quite well. Due to the long idle time the variance is enhanced, which has an effect on the relation between number of responses and time estimation. The theoretical assumptions would predict increasing duration estimation with increasing responses. That would mean that with increasing active time (30, 45, 55 seconds) the estimates would increase, because subjects process more patterns in the given interval of 60 seconds (mean numbers of responses: 39, 60, 71). But our experiment showed that due to the described variance the estimates are moderated by the idle time. In the case with 55 seconds active time the subjects showed the shortest median estimated time whereas in the case with 15 seconds idle time they showed the longest median estimated time. This effect is reproduced by the timer-module.

Discussion

The results show that our timer-module is able to reproduce retrospective time-duration estimations. The model data provide a very good fit for our task with a total task time of 60 seconds and different idle time portions. So far a model of McClain's (1983) experimental design is not implemented, but as the algorithm is able to reproduce her data, we assume that our timer-module would also provide a good fit.

In our model the number of responses in the D2-Drive increases with increasing task time, opposite to McClain's experiment where the number of responses remains rather low even in 120 seconds total task time. Due to this relation we have to point out, that for our model the estimation of different time-interval lengths is currently not proved.

We assume due to the model data that the computational model of retrospective time-estimation can predict the behavior of subjects in conditions of 60 seconds with idle times different to our investigated conditions.

The timer-module allows setting several reference-points one after another and provides estimations between successive reference-points as described earlier. It is not possible to get an estimation between non-successive reference-points by the timer-module automatically. Nevertheless a manual calculation of the estimate value is possible.

We have seen that idle time has a tremendous effect on variance in retrospective time-duration estimations. So far the timer-module provides just one estimate value and the observed variance is not represented, although we think that this would be helpful information.

Outlook

The empirical evidence for the proposed duration estimation algorithm has to be extended. Therefore follow-up experiments are already planed as well as some modifications of the underlying ACT-R model and the timer-module.

Literature just gives a fragmentary picture on questions related to contextual changes. We plan a meta-analysis to define contextual changes more precisely and to find competing approaches for the implementation of the underlying duration estimation algorithm.

Furthermore we want to integrate the observed variance induced through the idle time with a range index. That would offer a better understanding of the reproduced estimations.

In addition we would like to implement the possibility to question duration estimation over non-successive reference-points, that means to integrate successive estimations to one.

Another issue of our research is the integration of prospective time-duration estimation. In our opinion the combination of retrospective and prospective time-duration estimation might be a promising approach to explain duration estimation and its complex effects. Both methods should be integrated in cognitive architectures. This opens a wide range of new applications in the field of designing dynamic human machine systems.

Acknowledgments

This work was funded by grants of VolkswagenStiftung (research group User Modeling in Dynamic Systems) and Deutsche Forschungsgemeinschaft (research training group GRK 1013 prometei).

References

Anderson, J.R & Lebiere, C. (1998) *The atomic components of thought.* Mahwah, NJ: Erlbaum.

Block, R. A., & Reed, M. A. (1978). Remembered duration: Evidence for a contextual-change hypothesis. *Journal-of-Experimental-Psychology, Vol. 4, no. 6*, 656-665.

Block, R. A., & Zakay, D. (1997). Prospective and retrospective duration judgments: A meta-analytic review. *Psychonomic-Bulletin-and-Review, 4(2)*, 184-197.

Bothell, D. (2005). *ACT-R Software Framework Specification v 1.2.9*. Retrieved from svn://alba.psy.cmu .edu/usr/local/svnroot/actr6 [14.12.2005].

Brickenkamp, R. (2001). *Test d2 Aufmerksamkeits-Belastungs-Test. 9., überarbeitete und neu normierte Auflage*. Bern, Schweiz: Hogrefe Verlag.

Card, S. K., Moran, T. P. & Newell, A. (1980). The keystroke-level model for user performance time with interactive systems. *Communications of the ACM, Vol. 23, no. 7,* 396-410.

Hamacher N., Kraiss K.-F., Marrenbach J. (2002). Einsatz formaler Methoden zur Evaluierung der Gebrauchsfähigkeit interaktiver Geräte. *it + ti Informationstechnik und Technische Informatik , Vol. 44, no. 1*, 49-55.

Hicks, R. E., Miller, G.W., & Kinsbourne, M. (1976). Prospective and retrospective judgments of time as a function of amount of information processed. *American Journal of Psychology, 89,* 719-730.

Just, M.A., & Carpenter, P.A., (1992). A capacity theory of comprehension: Individual differences in working memory. *Psychological Review, 99(1)*, 122-149.

Laird, J., & Rosenbloom, P. (1996). The evolution of the Soar cognitive architecture. In Steier, D. M., & Mitchell, T. M. (Eds.), *Mind Matters: A tribute to Allen Newell*, (pp. 1-50). Mahwah, NJ: Erlbaum.

McClain, L. (1983). Interval estimation: Effect of processing demands on prospective and retrospective reports. *Perception and Psychophysics, 34(2),* 185-189.

Pape, N., Dzaack, J., Leuchter, S. & Urbas, L. (2005). How to integrate time-duration estimation in ACT-R/PM. *In Proceedings of the 12th Annual ACT-R Workshop*, Trieste, Italy.

Schulze-Kissing, D., van der Meer, E. & Urbas, L. (2004). A Psychological Analysis of Temporal Errors in Human-Machine-Systems. *Proceedings of the IFAC Symposium: Analysis, Design and Evaluation of Human-Machine-Systems. Atlanta, USA, 07.-09. September 2004.*

Urbas, L., Schulze-Kissing, D., Leuchter, S., Dzaack, J., Kiefer, J., Heinath, M. (2005). *Programmbeschreibung D2-Drive-Aufmerksamkeitstest*. Berlin: ZMMS.

Zakay, D., and Block, R. A. (2004). Prospective and retrospective duration judgments: an executive-control perspective. *Acta Neurobiol,* 64, 319-328.