A Preliminary ACT-R Model of a Continuous Motor Task

Michael D. Byrne*, Marcia K. O'Malley†, Melissa A. Gallagher*,
Sagar N. Purkayastha†, Nicole Howie*, and Joel C. Huegel°
*Department of Psychology, †Department of Mechanical Engineering and Materials Science, Rice University
°Department of Mechatronic and Electronic Engineering, ITESM, Guadalajara Campus, Guadalajara, Mexico

Cognitive architectures such as ACT-R and EPIC are being applied to human factors research problems with increasing frequency. However, it is unclear whether such systems can model continuous motor tasks that were once staples in the field but have since been largely displaced by more cognitively-oriented problems. Recent research on a challenging continuous motor control task has revealed interesting patterns in skill acquisition that appear compatible with the learning mechanisms present in ACT-R. However, what was not clear was whether ACT-R could model expert performance in a high-frequency motor control task. Unmodified, ACT-R could not. However, by making some small changes in ACT-R's motor system and capitalizing on ACT-R's ability to imagine visual objects, ACT-R was able to achieve expert-level performance in this task. Whether ACT-R will be able to mirror the skill acquisition data is still an open question.

INTRODUCTION

Over the last decade and a half, human factors has seen an increase in the use of modeling techniques that have their roots in computational cognitive architectures, e.g., ACT-R (Anderson, 2007) and EPIC (Kieras, et al., 2000). This is likely due at least in part to the increasing emphasis cognitive factors have had in human factors over the previous 30 or so years (see, for example, Byrne & Gray, 2003), but also to the increased ability of such architectures to "scale up" to tasks of relevance to human factors, such as graph comprehension (Peebles & Cheng, 2003) and driving (Salvucci, 2006). However, it is important when a field adopts new methodologies to consider whether those new methodologies are able to handle problem domains covered by older methodologies.

An example of such an area is manual control and tracking. Extensive work has been done on the topic of manual tracking including careful refinement of two models, the "crossover" model and the optimal control theory model, which together have been able to shed a great deal of light on human manual tracking in a variety of contexts. (Space prohibits a substantive treatment of these models; see Jagacinski & Flach, 2003, for an introduction.) What do cognitive architectures have to say about such domains? This kind of problem has received little attention from the cognitive architectures community; the most notable exceptions are the EPIC-based work on the Martin-Emerson and Wickens tracking/choice RT task (Kieras, et al., 2000) and Salvucci's (2006) ACT-R work on driving. Those are both important pieces of work but neither one has as its primary focus the motor control aspects of the task. This is hardly surprising, since motor control is not the primary focus of such architectures; they are not termed "cognitive" architectures for nothing.

However, there are tasks with strong motor control components that also appear to have interesting cognitive components. For the remainder of this paper, we will consider one such task, researched extensively by O'Malley and colleagues (O'Malley, et al., 2006; Li, Patoglu, & O'Malley, 2009; Huegel, et al., 2009).

The Task and the Phenomena

The task used in this research is a motor control task but not strictly a tracking task since the targets are stationary and the only movement is operator-generated. Nonetheless, it is a challenging task. The operator is seated at a computer display and grips a joystick, which controls a disk on a computer screen. The disk is coupled to a second disk, and the goal of the task is to move the controlled disk (termed the "tool") such that the coupled disk hits first one target, then the other. The disks are modeled as masses coupled by a damped spring. This is depicted in Figure 1.

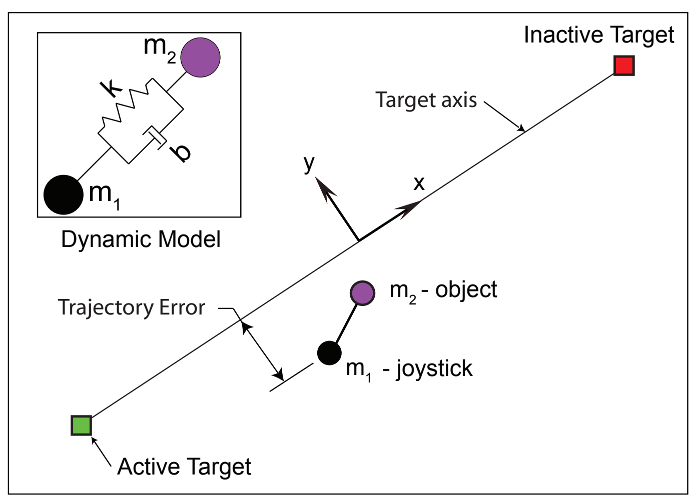

Figure 1. The task configuration. m_1 (termed the "tool") is controlled by a joystick; the goal is to alternate hitting the two targets with m_2 (termed the "disk").

The task is scored according to the number of target hits that the operator can score in a 20-second interval. This is a challenging task for many subjects. However, with practice, most (but not all) subjects become fairly proficient with the task and can generate slightly more than 1 target hit per second. One of the things that makes this task interesting is precisely the issue of learning. Some subjects start out poorly and improve only a modest amount across multiple experimental sessions. Still other subjects start out doing well and show a similar modest improvement, generating strong scores across all trials. Finally, a third group of subjects starts out doing poorly, but learns rapidly and ends up doing about as well as subjects who started out strong. Figure 2 presents data from Huegel (2009) showing this breakdown. High performers are defined as subjects whose initial hit count performance is more than one standard deviation above the mean. Low performers are defined as subjects whose final hit count performance is more than one standard deviation below the mean. The third group consists of all other subjects; they transition from performing like low performers into performing like high performers.

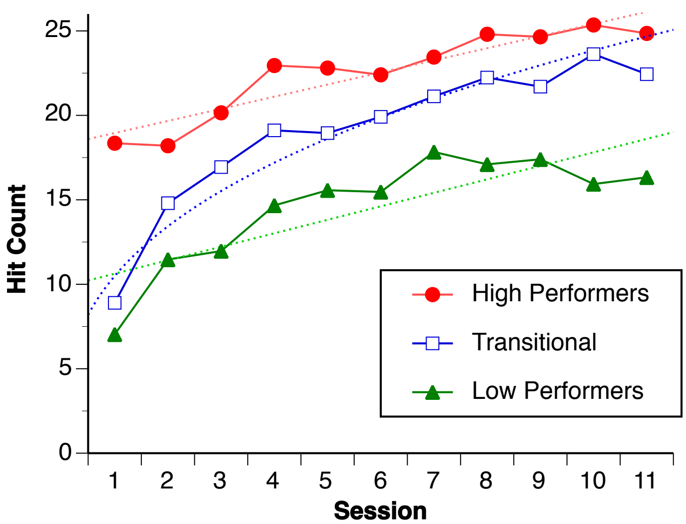

Figure 2. Mean hit count per trial across sessions for the three subject groups in the spring-target task.

This is an interesting learning domain because these curves have distinct shapes. This raises a number of important and as-yet unanswered research questions, such as:

- Can group membership be predicted in advance, perhaps by examining some other measure or with some other shorter test?
- What is it that the high performers know that enables them to perform well even early in learning?
- Can we understand what it is that the transitional subjects learn? If so, can that be taught to low performers to bring up their performance?

ACT-R is a general-purpose (i.e., multi-domain) computational theory of human cognition and performance which synthesizes research findings in psychology, computer science, and neuroscience and provides a natural platform for investigating questions about human performance. ACT-R is an attractive tool for addressing the questions raised by these data because of the nature of the learning mechanisms present in ACT-R (Anderson, 2007). The slow, nearly linear learning of both the high and low performers is characteristic of the kind of speedup learning produced by reinforcement learning, which is ACT-R's primary method for "tuning" procedural knowledge. (Reinforcement learning does not produce exclusively linear results, but often does.) The clearly non-linear learning exhibited by the third group, however, appears to be something qualitatively different. What ACT-R suggests is that those learners do, in fact, acquire some new declarative knowledge during the course of learning, and that this knowledge is then compiled into procedural knowledge, a process which generally produces the kind of super-linear learning exhibited by those subjects.

On the other hand, ACT-R may not be an ideal fit, since this kind of continuous motor task is not the kind of task that ACT-R has traditionally modeled. If ACT-R cannot even do the task as well as the high performers, there is little to be gained by trying to understand how it might learn to improve its own performance. Thus, our first question in addressing this research was: can ACT-R match the best human performance? At the outset, it was not at all clear that ACT-R could do so in such a task. Thus, the primary goal of this paper is to describe our efforts at matching an ACT-R model to the end-of-practice data from the top two groups, which we term "expert" performance.

METHODS

Movement Profiles for Experts

An important first step in modeling expert performance is understanding that performance at a detailed level. We are interested in constructing an ACT-R model that both produces high scores and achieves those scores in a way that is similar to how human experts do.

Fortunately, Huegel (2009) performed a comprehensive analysis of the movement characteristics of human experts on this task. What he discovered is that human experts do two things:

- Movement is almost exclusively along the target axis. That is, experts do not move in circles or ellipses (as many subjects do, especially early in learning), but move more simply back-and-forth along the target axis.
- Fourier analysis of expert movements indicate high consistency of movement frequency; in particular, experts tend to match or slightly exceed the natural frequency of the system. Their motion is highly regular, point-to-point and back over the same time span. Experts do not simply move as

fast as they can, but tend to keep their movements in time with the natural frequency of the system.

These two variables explained most of the variance in the number of target hits between individuals and between sessions, and were only weakly correlated with each other, indicating that these are two key but separable components of skill in this task.

Discrete Movements

The first fundamental problem this task poses for ACT-R is that it, like many other such tasks, has the appearance of being continuous. The coupled disk is in constant motion and the tool is in motion most of the time. However, ACT-R has no capability for continuous motion. ACT-R's manual motor system is a limited abstraction designed to model a fairly limited range of motor performance: essentially simple typing and mouse movements targeted at visible objects on a computer screen. Furthermore, ACT-R's movements are entirely linear, start point to finish point movements (though there is noise in the finish point).

This task provides some opportunity to test the veracity of such approximations. While being restricted to straight-line movements is clearly inadequate for many tasks (e.g., gesturing), expert performance in this task happens to consist of almost entirely straight-line movement. As Craik observed more than 50 years ago (Craik, 1947), even in continuous tracking tasks, human movement is often not actually continuous at all, but made up of a series of intermittent small movements. ACT-R instantiates this idea.

However, this is not to say that we could use the ACT-R motor system unmodified; several modifications were necessary. First, ACT-R's movement output is quite coarse. The behavior of ACT-R when making any kind of aimed movement was to compute the duration of movement (via Fitts's Law) and simply alter the location of the hand/finger/cursor once that amount of simulated time had elapsed. This does not work for simulated environments such as this one; this essentially generates instantaneous movement which wreaks havoc on the physics-based simulation. ACT-R could previously be set to output movements in 50 ms increments, but even that is far too coarse for this environment. Thus, we modified ACT-R to allow for more flexible updates to the output position. For the model runs reported here, we updated the motor output location (the tool position) every 3 ms. Note that this was not a change to the cycle time for either cognition or motor modules, but simply a change in how often the system updates the simulated position of the moving hand.

We also had to modify the velocity profile for the movement. ACT-R previously assumed constant velocity for the entire duration of the movement until the movement was complete, at which time the velocity was set to zero. While this is perfectly adequate for many circumstances, it again caused no end of problems for the physics-based simulated environment, because the acceleration of the tool jumped from zero to a constant instantaneously at the beginning of the movement and then from that constant back to zero at the end of the movement. These huge accelerations and decelerations imparted a great deal of force to the coupled disk, causing excessive motion. The solution to this was to alter the function controlling ACT-R's output position over time. The simple linear function was replaced with a "minimum jerk" movement profile (Hogan, 1984). This profile minimizes the derivative of acceleration throughout the movement, resulting in a very smooth movement. This is almost certainly a little too smooth relative to actual human aimed movement (cf. Jagacinski & Flach, 2003) which tends to involve many very small velocity corrections over the course of the movement, but it has the advantage of being not too far off from real human movement profiles and is fast and easy to compute.

The final, and most theoretically interesting, problem here is that all the movements in ACT-R other than some very simple ballistic moves (e.g., punch the key directly below a finger) are aimed movements. That is, they require a visual target for the movement. This target is required in order to compute a target width for the computation of index of difficulty in Fitts's Law. However, this creates a problem in that the only visual objects on the display are the tool, the disk, and the two actual targets, and none of those are appropriate as movement targets.

Fortunately, we were able to co-opt an existing piece of ACT-R functionality to solve this problem. ACT-R has the ability to form new representations on the fly through a system termed the "imaginal" system. This imaginal system is typically used by ACT-R models to store a representation of some part of the current problem, such as a mental marker for a carry operation in multi-column arithmetic. In our model, ACT-R uses this same system, but imagines a visual representation—a virtual movement target—and the spatial properties (location, size) of that imagined target can be passed along to the motor system just as if an actual visual object were the target. In this way, ACT-R can move to virtual "objects" that do not actually exist on the display. Using the imaginal system takes some time, however, as the representation must be constructed in the imaginal system, and that takes some 200 ms.

RESULTS

With these pieces in place, it is possible to construct a relatively straightforward ACT-R model of expert performers in the spring-target task. Essentially, the model generates (imagines) a virtual object in the upper right, shifts visual attention to that object, and moves the tool to it. Following that, it imagines a virtual target in the lower left, shifts visual attention to that object, then moves the tool to it. The cycle then repeats.

This cycle repeats at approximately 1 Hz on average for both the model and for human subjects. There are two sources of stochasticity in the model. First, the durations of all the primitive operations are noisy (drawn from a rectangular distribution), which generates noise in both the timing of the model and the velocity of the tool. Second, the actual endpoint of each movement is non-deterministic. The targeted point is the center of a two-dimensional Gaussian distribution, with the width of that distribution being proportional to the target object's width; the final move endpoint is drawn from this Gaussian. These are standard noise settings in ACT-R; no parameter tuning was done here to improve data fits.

Over 200 runs of the model it produces a mean hit count of 23.96 hits per 20-second trial; the largest mean for the human subjects (high performers in session 10; see Figure 2) was 25.35; the model is off by about 5%. We had to set the values of two free parameters in order to produce this performance, but both of those parameters were set in principled ways and were not set by empirical data-fitting.

The first parameter is the location of the imagined or virtual target objects. (Technically there are two locations but these were set symmetrically along the target axis.) These locations were set based on the physics of the resonant frequency of the system. The location of the virtual target objects is determined based on the dynamics of the sprung mass system. The ratio of the output amplitude to the input amplitude depends on the excitation frequency and the damped properties of the system. The stiffness and damping parameters are chosen such that the behavior of the system is slightly underdamped. For underdamped behavior, when the system is excited in a sinusoidal manner at or near its natural frequency (which is a function of the mass of the tool and the stiffness of the spring), the output amplitude will be greater than the input amplitude. Therefore, the best excitation strategy for this system is to move the disk between two points that are located at the precise distance from the origin for which the motion of the tool will just reach the targets. For example, if the output amplitude to input amplitude ratio at the natural frequency is 2, then the virtual targets should be half-way between the origin and the actual targets. Slight overshoot of the virtual targets will not adversely affect performance in terms of hit count, but the increased travel will require greater energy input in order to keep the input excitation near the natural frequency, or will result in excitation slightly below the natural frequency of the system.

There is one small complication here. The very first movement made by human subjects is larger than all subsequent movements. The system starts with all objects stationary, so getting the system ramped up to its natural frequency takes some time. In order to impart a higher early velocity, subjects make one initial move that is larger than other movements. There are substantial individual differences in the exact location of this initial movement; we approximated this by visually examining the data from a few expert subjects. This is, in a sense, a free parameter, but the ultimate performance of the model is not particularly sensitive to the exact location of the target for the first movement, only that the first movement is large enough to impart some additional velocity.

The other parameter was the size of the virtual target objects. Both the tool and disk are circles with 20-pixel diameters so the natural choice would be to use a virtual target that is the same effective size as the visible objects on the display. By default, visual objects in ACT-R are square or rectangular and the effective width of an object is the length of the chord running through the center of the object based on the line from the starting position to the object center. Since the virtual object was set as a square and target axis is on a 45° angle, the approach chord is 1.41 times larger than the height/width of the object. Thus, the target was set to be 14 pixels wide, giving it an effective width of 20 pixels, which is the same as the size of the controlled objects.

It is interesting to note that we did experiment with different sizes for the virtual targets, but found 14 pixels to be the optimal setting. Using a larger target sped up the system (wider targets have a lower index of difficulty in Fitts's law and therefore result in more rapid movements), but introduced too much off-axis error and would cause the model to occasionally miss a target, hurting performance. Using a smaller target slowed down the movement such that the model could not achieve the speed necessary to maintain the resonant frequency of the system.

There are some subtleties in the data that the model does not address. For example, the noise added by ACT-R to the movement endpoints uses the same variance for on-axis error as for off-axis error. This appears not to be true for human subjects; off-axis error appears to be slightly smaller than on-axis error. This may contribute to the model's slight underperformance relative to the best experts; perhaps the virtual targets could be slightly larger, and therefore the moves slightly faster, if the off-axis noise were reduced relative to the on-axis noise. Overshoot of the virtual targets has a limited impact on score since the tool still passes through the target (see above), as opposed to deviation from the target axis which is more likely to lead to missed target hits.

DISCUSSION

What the model demonstrates is that it is possible to get ACT-R to perform approximately as well as human experts at a continuous, dynamic motor task. This is encouraging in that it leads to the possibility that ACT-R could learn to become an expert at this task, which may shed light on how to train human subjects on the task. This is an interesting issue in part because it is not clear how to train subjects on this task; past attempts using visual guidance and haptic feedback have not met with much success (Li, Patoglu, & O'Malley, 2009; Li, et al., 2009).

It is not entirely clear that ACT-R can learn this task without outside assistance. The model presented here "knows" both a particular strategy (oscillate between two imagined targets) as well as as a set of parameter values (the size and locations of the virtual targets). We are currently exploring methods by which the model could learn the size and locations of the virtual targets through simple reinforcement learning. If this is successful, we will be able to claim, at best, that we have a model of the high performers.

However, how the other two groups of subjects could be modeled is not as clear. An examination of the low performers' movement profiles indicates it is likely that they are using an entirely different strategy since they tend to make oval-shaped movements, even after many sessions, rather than adopting a strict back-and-forth strategy. This is challenging for ACT-R since it does not presently handle curved movement paths well, though extending it to do so is not impossible.

The particularly difficult group is the the third group. Subjects in this group appear to transition from oval-shaped movements to the back-and-forth strategy exhibited by experts. In some sense having ACT-R learn this is trivial; if ACT-R is initially provided both strategies it can certainly learn to prefer the better strategy over time, and we would certainly explore such an approach.

However, this is not entirely satisfying as an explanation on two fronts. First, from what source would ACT-R get these two strategies in the first place? Second, if ACT-R can learn to prefer the better one, why don't all the actual human subjects? Unless, of course, this is exactly what separates the low performers from the fast learners; the fast learners are simply subjects who have both strategies available. Even still, this leaves as an open research question how subjects would have acquired these strategies to begin with. Inducing entirely new strategies has not traditionally been ACT-R's strength, though there are some recent ACT-R models which are able to do essentially this (Anderson, 2007). This kind of induction is a very different kind of learning than the kind of simple tuning associated with later motor learning and, to the best of our knowledge, has not been systematically explored in the context of a difficult motor learning problem.

It may be the case that what humans learn when transitioning is an internal model of the dynamics of the system, much like the Kalman filters found in optimal control models (again, see Jagacinski & Flach, 2003, for an overview). ACT-R does not, at present, have dynamic motor system models, so this is a rich area for future research.

That research has important potential long-term benefits. Difficult motor domains are precisely the kinds of domains where a deeper understanding of the learning process is valuable. Training in such domains is traditionally both difficult and time-consuming. Our ultimate goal in trying to understand skill acquisition in this task is to be able to provide training support—perhaps haptic or visual guidance, perhaps instructions—to help the low performers transition into high performers. We will then to attempt to generalize that to other complex motor domains such as mediated surgery or remote piloting. This research is the first step down such a path.

ACKNOWLEDGEMENTS

This research was supported by the National Science Foundation under grant #IIS-0812569. The views and conclusions contained herein are those of the authors and should not be interpreted as representing the official policies or endorsements, either expressed or implied, of the NSF, the U.S. Government, or any other organization.

REFERENCES

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

Byrne, M. D., & Gray, W. D. (2003). Returning human factors to an engineering discipline: Expanding the science base through a new generation of quantitative methods—preface to the special section. *Human Factors*, *45*, 1-4.

Craik, K. J. W. (1947). Theory of the human operator in control systems. I. The operator as an engineering system. *British Journal of Psychology, 38*, 56–71.

Hogan, N. (1984). An organizing principle for a class of voluntary movements. *Journal of Neuroscience, 4*, 2745–2754.

Huegel, J. C. (2009). Progressive Haptic Guidance for a Dynamic Task in a Virtual Training Environment. Doctoral Dissertation, Rice University, Houston, TX.

Huegel, J. C., Celik, O., Israr, A., & O'Malley, M. K. (2009). Expertise-based performance measures in a virtual training environment. *Presence: Teleoperators and Virtual Environments, 18*(6), 449–467.

Jagacinski, R. J., & Flach, J. M. (2003). *Control theory for humans: Quantitative approaches to modeling performance*. Mahwah, NJ: Erlbaum.

Kieras, D. E., Meyer, D. E., Ballas, J. A., & Lauber, E. J. (2000). Modern computational perspectives on executive mental processes and cognitive control. Where to from here? In S. Monsell & J. Driver (Eds.), *Control of cognitive processes: Attention and Performance XVIII* (pp. 681–712). Cambridge, MA: MIT Press.

Li, Y., Huegel, J. C., Patoglu, V., & O'Malley, M. K. (2009). Progressive shared control for training in virtual environments. In *Proceedings of the International Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems and the Third Joint World Haptics Conference (HAPTICS)* (pp. 332–339).

Li, Y., Patoglu, V., & O'Malley, M. K. (2009). Negative efficacy of fixed gain error reducing shared control for training in virtual environments. *ACM Transactions on Applied Perception, 6*, 1–21.

O'Malley, M. K., Gupta, A., Gen, M., & Li, Y. (2006). Shared control in haptic systems for performance enhancement and training. *ASME Journal of Dynamic Systems, Measurement and Control, 128*, 75–85.

Peebles, D., & Cheng, P. C.-H. (2003). Modeling the effect of task and graphical representation on response latency in a graph reading task. *Human Factors, 45*, 28-45.

Salvucci, D. D. (2006). Modeling driver behavior in a cognitive architecture. *Human Factors, 48*, 362–380.