Addition as Interactive Problem Solving

Hansjörg Neth (NethH@Cardiff.ac.uk)
School of Psychology, Cardiff University
Cardiff CF10 3YG, Wales, United Kingdom

Stephen J. Payne (PayneS@Cardiff.ac.uk)
School of Psychology, Cardiff University
Cardiff CF10 3YG, Wales, United Kingdom

Abstract

Successful problem solving depends on a dynamic interplay of resources between agent, task, and task environment. To illuminate these interactions we studied how participants added a series of single-digit numbers presented on a computer screen. We distinguished between four different user interfaces, each implementing a different mode of interaction with the displayed addends: look only, point, mark, and move. By collecting and analysing complete interaction protocols we were able to integrate overall performance measures with fine-grained behavioural process data on the strategies engendered by the different user interfaces. We discovered reliable differences in the chosen sequences of addends, which can be understood in terms of the cost-benefit structures provided by the interactive resources of the user interfaces.

Introduction

Successful problem solving is an embedded and embodied process and crucially depends on a dynamic interplay of resources and constraints between agent, task, and task environment.

The importance of feedback loops, wherein actions on the world provide new information to the problem solver, was recognized in the earliest cognitive accounts of human problem solving (e.g. Miller, Galanter and Pribram, 1960; Newell and Simon, 1972). Yet until relatively recently, the interactive properties of the task environment have seldom been the focus of attention. Thus, the traditional literature on problem solving has been concerned primarily with planning, search strategies and heuristics (see e.g. Mayer, 1992, for an overview).

Recently, however, it has become increasingly clear to many investigators that interactions between mental processes and external objects play a crucial role in human problem solving. This interactive perspective has led to recent analyses of, for example: the importance of constraints provided by the properties of external representations (Larkin and Simon, 1987; Zhang and Norman, 1994; Zhang, 1997); the role of the display as a resource in human-computer interaction (Payne, 1991; Monk, 1998; Gray and Fu, 2001); the effect of the cost of implementing operators on the interplay between planning and action (O'Hara and Payne, 1998).

In this article, we extend this general approach to investigate the way in which the nature of available interactions in the task environment determines the discovery and use of strategies in a rather simple problem solving task: adding a series of numbers.

This work builds on the empirical work of Kirsh and colleagues (1995a, 1995b, Kirsh & Maglio, 1994), who, in a series of empirical studies, have shown that problem solvers often spontaneously manipulate the external world in order to reduce cognitive load.

In studying the interactive video game "Tetris", Kirsh and Maglio (1994) showed that expert players physically rotated falling pieces more than was required by their goal orientation. Kirsh (1995) demonstrated that people were reliably faster and more accurate at counting coins when they were allowed to move the coins around as they counted. Similarly, Maglio et al. (1999) found that people generated more anagrams when they were allowed to rearrange Scrabble tiles as they worked.

The experiment reported in the current article exemplifies an empirical approach that has three characteristics:

First, we study a problem solving task in which the atomic components are relatively simple and well understood, so that strategy differences (as well as outcome differences) may be easier to observe and explain.

Second, we design several user interfaces to the same problem, allowing subtle manipulations to the interactive resources that are available to problem solvers. This enables a more refined investigation of the relationship between resources and strategies.

Third, we independently manipulate problem complexity, so that we can assess relations between problem characteristics and interactive resources.

Interaction in Addition

Consider a simple serial addition like 1+2+9+7 presented, as here, linearly on a visual display. As with many cognitive tasks, we could solve this entirely "in our heads". But this does not warrant the conclusion that environmental interactions cannot play important roles.

Despite the linear presentation format, the law of commutativity allows us to add the four addends in any of 4!=24 different orders. Whilst all potential solution paths result in a total sum of 19 their cognitive demands may vary considerably. Within the context of this study, two sequences are of particular interest: By first adding 1 plus 9 before adding 2 and 7 to the result, one could exploit the fact that within the arabic base 10 number system the two addends 1 and 9 form what we call a *pair*, i.e., they add up to the next bigger unit, a "round number". Likewise, someone might first add 1+2 but then spot 7 to make an intermediate sum of 10 before adding 9. In this case, the single addend 7 *complements* the current intermediate sum 3 to make a round number.

Both strategies exploit the same rationale: Two numbers, which add up to a round number are easy to add, and, when adding series of numbers, it is easier to add another number to round intermediate sums. The difference between a pair and a complement strategy is that pairs combine two external addends, whereas complements combine an internal intermediate sum with an external number.

However, both strategies come at a cost. As neither the pair nor the complement in the above example is available with adjacent elements of the linear left to right sequence, their detection requires visually searching ahead through the problem display, as well as some way of keeping track of used and skipped numbers. Thus, the use of a pair and complement strategy facilitates calculation at the expense of other resources. We hypothesize that the specific structure of this trade-off depends on the triad of factors noted above: The skill and memory capacity of the problem solver, the difficulty of the problem, and the availability of interactive resources.

Pilot Study In a pilot study we observed that the ability to rearrange or manipulate numbers on paper cards improved performance in simple addition. Furthermore, the availability of a pencil encouraged participants to use both the pair and the complement strategy, particularly with increasing problem difficulty and when numbers were presented in a 2-dimensional array. However, particular ways of using the pencil varied greatly (and included pointing, marking, copying, as well as recording intermediate sums), obstructing precise analysis of the underlying effects. Consequently, in the current study, we use computer interfaces to isolate specific interactive resources.

Method

Participants' interactions with the problem of adding numbers were operationalized as mouse actions and visual feedback on a standard computer interface. Four different *interactive modes* were distinguished:

1. *Look only*: Numbers had to be added without being able to point at them, as the mouse cursor was disabled during stimulus presentation.
2. *Point*: The mouse cursor was enabled and participants were instructed to click on numbers when adding them. When a number was clicked, a brief tone provided auditory feedback.
3. *Mark*: Mouse pointer and instructions were exactly as in 2. However, when a number was clicked, it also changed its colour from dark red to grey, thereby visually marking numbers that had been processed.
4. *Move*: Numbers could be moved on the screen using a drag-and-drop procedure.

Table 1: Examples of linear stimulus lists allowing for pairs, complements, or neither at positions x_1-x_2, y_1-y_2, and z_1-z_2.

Type	Stimulus list			Sum
Pair list:	4 3 9 7	8 6 5 4	2 1 5 9	63
Complement list:	3 1 8 6	5 3 9 4	5 7 2 9	62
Neutral list:	9 4 5 8	9 6 3 2	1 5 7 2	61
Structure:	$a\ x_1\ b\ x_2$	$c\ y_1\ d\ y_2$	$e\ z_1\ f\ z_2$	

Materials A total of 72 lists of 4, 8, or 12 single-digit numbers were generated by a Prolog program. Each list consisted of one, two, or three building blocks of the form ax_1bx_2. Three types of linear lists were distinguished: For *pair lists* x_1 and x_2 added up to 10 and the list allowed for no complements within a lookahead span of three digits. Analogically, for *complement lists* the value of x_2 plus the intermediate sum at x_1 resulted in a round number, and the list contained no pairs within a lookahead span of three digits. *Neutral lists* allowed for neither pairs nor complements within the same lookahead span. Note that none of the linear lists contained any adjacent pairs or complements (see Table 1 for some examples of stimuli). In contrast to linear lists, the elements of *spatially distributed lists* were scattered pseudo-randomly over the screen. Lists within each level of the list-length and -type factors were matched for their sums and number of possible pairs.

Stimuli presentation and data collection were controlled by a MS Windows Visual Basic program. All stimuli were displayed on a 17" computer screen using a 20pt Arial bold font of dark red colour against a white background.

Design The experiment used a mixed design, with *interactive mode* as a between-subjects manipulation and *list length* and *list type* as within-subjects factors.

Procedure Forty-four Psychology undergraduates (with a mean age of 20.3 years) took part in the experiment to receive course credit and were randomly assigned to one of the four interactive modes.

After the completion of four practice trials and a letter task to familiarize participants with their respective interactive resources, participants were instructed to add as fast as they could without making any errors. For each trial, participants pressed a button when they had added all numbers and then entered the result on another screen using a mouse-operated number pad.

Since erroneous trials were repeated at the end of the randomized sequence of trials, the experiment continued until the participant correctly added all 36 different lists (i.e. three different lists of each of three lengths and four types). On average, participants completed the experiment within 25 minutes.

Predictions The first and most basic prediction is that participants will benefit from interactive resources. In particular, following the findings of Kirsh and colleagues (1994, 1995a, 1995b), we predict that the *move* condition will elicit better performance than the *look only* condition. We also make a specific prediction concerning the comparison between the *point* and *mark* conditions. Because the latter provides an external memory for already-processed addends, it reduces the cognitive costs associated with the more sophisticated strategies of exploiting pairs and complements. Thus we predict more use of these strategies in the *mark* condition than the *point* condition, and more efficient performance as a result.

Results

Analyses of time and accuracy for the practice trials showed no differences between experimental groups at the pre-test stage. In the following report, we first focus on performance measures before considering more detailed process characteristics.

Performance

Accuracy The overall rate of errors was 13.87%. A one-way between subjects ANOVA confirmed that the number of erroneous trials in the four experimental groups differed between interactive modes [$F(3,40)=5.8$, $p=.002$, $MSE=15.2$, see Table 2 for descriptive data].

Planned comparisons revealed that participants in the *look only* condition indeed had significantly more erroneous trials than those who could *move* numbers ($t=3.28$, df=40, $p=.002$). Likewise, participants in the *point* condition made significantly more errors than those in the *mark* condition ($t=2.57$, df=40, $p=.014$). Thus, both of our specific predictions were confirmed for accuracy data.

Latency As differences in accuracy could be due to a speed-accuracy trade-off error rates and response latencies must be considered in parallel. Since the total time participants spent on the experiment is trivially longer when they made more errors we divided it by the actual number of trials for each participant to obtain an overall measure of average time per trial.

The comparison of accuracy and latency data (see Table 2) shows that no speed-accuracy trade-off contaminated the between-groups comparisons: In addition to being less accurate, participants in the *point* condition were significantly slower than those in the *mark* condition ($t=3.03$, df=40, $p=.004$), whereas the three other groups did not differ with respect to overall latency (Tukey pairwise comparisons).

Table 2: Mean number of erroneous trials and latencies per trial in seconds (and their respective standard deviations). Each interactive mode included 10 participants.

	Interactive mode			
	look only	*point*	*mark*	*move*
Errors:	8.6 (5.9)	7.8 (3.8)	3.6 (2.6)	3.2 (2.3)
Latencies:	13.8 (2.9)	17.8 (4.4)	13.3 (3.2)	12.6 (3.6)

Table 3: Mean latencies of correct solutions in seconds (a) by list length (each cell contains 132 data points) and (b) by list type (each cell contains 99 data points).

		Interactive mode			
		look only	*point*	*mark*	*move*
(a) length	4:	4.7	7.2	6.9	4.7
	8:	12.8	16.3	12.9	11.3
	12:	21.8	27.4	19.0	20.4
(b) type	pair:	13.8	18.0	12.4	11.5
	complement:	13.6	16.9	12.7	12.5
	neutral:	13.2	17.8	12.5	12.0
	spatial:	11.8	15.2	14.0	12.5

Moderating Factors

So far, we have shown that overall performance measures varied across different interactive modes. However, this standard assessment of performance based on error rates and latencies does not distinguish between different task characteristics and thus cannot uncover potential interactions between tasks and interactive resources. To analyze how performance is modulated by problem features we now qualify the global between-subjects effects by the factors of list length and type.

Accuracy The effects of list length on the frequency of errors are as expected and consistent for all interactive modes: The longer the list, the more likely participants were to add it incorrectly. Also, it made no difference to the average error rate whether a stimulus was a pair, complement, neutral, or spatially distributed list.

Since any error in calculation could effectively alter the type and length of a list, all subsequent analyses examining the effects of list length and type will be based on correct trials only.

Latency A mixed ANOVA using a 4x3x4 design was conducted to assess the effects of interactive mode, list length, and list type. Apart from significant main effects of interactive mode [$F(3,40)=4.9$, $p=.005$, $MSE=125.8$] and list length [$F(2,80)=511.6$, $p<.001$, $MSE=22.96$] it yielded significant interactions between interactive mode and list length [$F(6,80)=5.6$, $p=.001$, $MSE=23.0$] and interactive mode and list type [$F(9,120)=5.3$, $p<.001$, $MSE=6.1$].

To interpret the results of subsequent simple main effects the mean latencies are shown in Table 3. Unsurprisingly, the time needed to add a list increased for all interactive conditions as the lists' length increased (see columns of Table 3a). However, the slope of this increase was much steeper in the *point* condition. For lists of four numbers, participants in the *look only* and *move* conditions were faster than the two other groups.

Of the eight possible simple main effects for the rows and columns of Table 3b five are significant. However, the *absence* of significant differences in three cases is more instructive: For spatially distributed lists the effects of different interactive modes levelled out [$F_{A@b4}(3,40)=2.0$, $p=.124$], which is due to the participants in the *look only* and *point* conditions being slightly faster than for other list types. Likewise, the differences between the mean latencies in the *mark* and *move* conditions for different list types failed to reach statistical significance [$F_{B@a3}(3,129)=2.3$, $p=.076$; $F_{B@a4}(3,129)=1.0$, $p=.402$], suggesting that the ability to mark and move numbers allowed participants of the corresponding groups to somehow transcend the linear and spatial constraints imposed by different list types.

Strategies

Having established that there *are* differences in performance we have to explain their genesis. We will attempt this by addressing strategy differences between groups which are reflected by features of the actual problem solving process. For this purpose, participants' cursor movements and mouse clicks in the *point*, *mark* and *move* conditions provided a rich source of fine-grained process data.

Mouse Moves per Trial When analyzing mouse cursor data, we use the term "move" to signify the physical movement from a number x_1 to a different number x_2. As each number has both value and location, moves can be characterized in terms of their distance and type, i.e., neutral, complement, pair, and triple. (In analogy to pairs, we defined a *triple* as three consecutive addends with a sum of 10.)

To obtain a measure of the amount of activity on each trial we computed the total sum of distances of all consecutive moves for each trial. A mixed 3x4x4 ANOVA on the total distance of moves per trial yielded a significant interaction between the two within-subjects factors list type and length [$F(6,180)=4.0$, $p=.010$] as well as a significant interaction between interactive mode and list length [$F(4,60)=3.5$, $p=.037$]. Whereas the first interaction merely reflects stimulus characteristics (e.g. that longer and spatially distributed lists afford longer moves) the second illustrates the modulation of moves by different interactive modes and lengths (see Table 4).

Table 4: Mean distances of cursor movements per trial in twips (1 twip = 0.05 pt = 0.01764 mm). Each cell contains 132 data points.

		Interactive mode		
		point	*mark*	*move*
length	4:	7 100	7 492	404
	8:	18 281	22 363	18 176
	12:	24 972	40 650	38 994
Total:		16 784	23 502	19 191

Table 5: Mean frequency (and standard deviations) of pairs, complements, triples, and neutral additions. Each cell summarizes data from 396 correct trials.

	Type of Move			
	Pair	*Compl.*	*Triple*	*Neutral*
point:	29.3 (14.0)	6.9 (3.2)	10.2 (7.5)	205.3 (14.7)
mark:	65.3 (38.3)	2.9 (3.2)	7.5 (2.8)	175.5 (10.3)

Whilst the increase in average move distances with longer lists was to be expected, it is notable that the slope of this increase is much steeper for the *mark* and *move* conditions. However, simple effect tests for the rows of Table 4 yielded a significant value only for lists of four numbers [$F_{A@b1}(2,30)=240.8$, $p<001$]. The lower value of the *move* condition at this length suggests why the corresponding latency was identical to the *look only* condition (see first line of Table 3a): Participants mostly chose *not* to move anything when adding short lists, but made use of their interactive potential when adding longer lists.

As the distances of moves in the *mark* and *move* conditions did not significantly exceed those in the *point* condition, activity per se cannot account for the reported differences in performance. To further illuminate potential strategy differences between groups, we have to consider process data on a within-trial level.

Choice of Next Addend At every non-last number within a stimulus participants faced the potential choice of which number to add next. We now examine the type of these choices and the corresponding move distances for the *point* and *mark* conditions, who had identical instructions (and differed only by the colour change of clicked numbers in the latter group) and both provided data on the complete paths of the chosen sequence of addends.

Table 5 contains the mean frequency of pairs, complements, triples and neutral moves chosen within correct trials. To appreciate the overall frequency of non-neutral moves we have to bear in mind that a participant had to select one or two neutral addends before being able to reach a round number and that none of the linear stimulus lists contained an adjacent pair, triple, or complement.

Pairs were the most frequent type of "rounding" move for both groups. T-tests for independent samples showed that participants in the *mark* condition chose more pairs [$t(12.6)=2.9$, $p=.012$] and fewer complements [$t(20)=2.9$, $p=.009$] than those in the *point* condition, whereas the number of triples did not significantly differ between groups [$t(14.9)=1.4$, $p=.181$].

When adjusting the frequency data for the number of *possible* pairs and complements on each particular trial, we found that participants in the *mark* condition in fact chose to add 62.3% of all possible pairs and 2.3% of all possible complements, whereas participants in the *point* condition chose 31.4% of all possible pairs and 5.8% of the possible complements. As pair and complement strategies compete for the same addends, it is likely that the increase in complements for the *point* condition is a mere by-product of the more persistent selection of pairs in the *mark* condition.

As with the performance measures above, the differential effects of move choices were modulated by the task characteristics of list length and type. Specifically, participants in the *mark* condition predominantely pursued pairs regardless of list length and type, whereas those in the *point* group only used pairs when stimuli were short or spatially distributed.

Distance of Next Addend Additional support for the special attractiveness of pairs can be obtained when considering move distance data at the within-trial resolution.

When choosing which number to add next, participants had to balance the costs and benefits associated with the numerical value and the physical distance of each addend. If our main hypothesis about interactive problem solving applies on this micro-level, how far someone ventures in order to select a specific next number ought to vary as a function of interactive resources and number value.

Because the physical distance of moves varies trivially as a function of list length and layout, we determined how many physically closer numbers a participant skipped on each move in order to choose the next addend. By dividing the number of moves to the physically nearest unprocessed number by the total number of moves for each trial we gained a "proximity index". Its value represents the percentage of moves to the closest number per trial and ranges from 100% (indicating that the closest neighbour was always selected) to $(n\text{-}1)^{-1}\%$ (as at least one of the n-1 moves within a stimulus of n addends leads to a next number). The average proximity index for the *point* condition was found to be 71%, compared to a value of 61% in the *mark* condition [$t(782.3)=5.9$, $p<.001$], which indicates that marking led to a decreased likelihood to select the nearest neighbour.

To answer the question *why* participants prioritized spatially more remote addends in the other 29% or even 39% of all cases, we have to combine data on move distances and types. To quantify the price of spatial relocation a participant was willing to pay in order to make a particular type of move, we counted the number of physically closer numbers skipped for each move. Average scores of 2.20 for pairs, 1.12 for complements and 0.53 for neutral moves indicate that, to reach a pair, participants skipped about twice as many numbers than to reach a complement, whose selection still led participants to ignore about twice as many closer numbers than a neutral addend.

Moving Pairs Is there any evidence that the preference for a pair strategy generalized to the *move* condition?

Because the interactive mode of this group differed from the *look only* and *mark* conditions in that it permitted the problem solver the freedom *not* to interact, we lack the data on complete sequences of addends. However, as we observed many participants of this group either re-arranging numbers pairwise on the screen or positioning one addend of a pair physically close to the other, we computed the total distance between all possible pairs at the beginning and end of each trial. Since a mere decrease of distances between pairs could also be caused by someone moving *all* items closer together, we divided the pre- and post-trial distances between all possible pairs by the corresponding sums of distances of all possible non-pairs. A significant decrease of this ratio from initially 0.14 to 0.10 [$t(395)=12.2$, $p<.001$] allows the conclusion that pairs were moved closer towards each other than non-pairs.

Discussion

The experimental manipulation of interactive resources resulted in reliable differences in performance, which were systematically modulated by task characteristics.

Participants in the *look only* condition did well when adding short lists, but became unreliable as the number of addends increased. A similar error rate and even more pronounced increase in latencies to add long lists showed the participants in the *point* condition to be at an even greater disadvantage—presumably because they paid the additional price for interacting (clicking) without receiving the benefit of marking. As both groups had to mentally keep track of the numbers added, their strategies were more conservative and reflected specific stimulus characteristics.

In contrast, members of the *mark* and *move* groups exploited their interactive resouces to transcend the constraints imposed by stimulus and task characteristics and actively implemented a facilitative pair strategy. Their significantly faster and more reliable performance emerged as a consequence of systematic differences on a behavioural micro-level.

This finding of spontaneous adaptation to the structure of costs and benefits at the user interface supports recent attempts to describe interactive behaviour within a rational analysis framework (O'Hara and Payne, 1998; Gray and Fu, 2001). In Gray and Fu's study, a subtle increase to the cost of external information (an eye-movement or a single mouse-click) led to users of a simulated VCR relying on imperfect memory. In our study a relatively subtle change to the information display reduces internal memory load and thus enables a more sophisticated strategy for ordering addends. What is important about studies like these is not so much that small changes to the task environment can produce reliable shifts in behaviour but that an analysis of the interactions between physical and cognitve costs and benefits can predict and explain the particular behaviours that emerge.

In the current experiment, the additional resources provided by the more powerful interactive modes were all available relative cheaply (as are the so called "epistemic" and "complementary actions"—like rotating objects or moving coins—in the studies by Kirsh, as cited above). What happens when complementary actions become more expensive, in terms of time or mental effort? Even in the current study, the results from the *move* condition suggest that disuse of interactive resources can sometimes be adaptive. In future work, we propose to investigate such questions by directly manipulating costs, following the methodology of O'Hara and Payne (1998), and by asking participants to explicitly choose between modes of interaction, using the choice-no-choice paradigm of Siegler and Lemaire (1997).

In this experiment, participants spontaneously, and almost instantly adopted a strategy which was tuned to their interactive resources. This contrasts with findings that people often are very inflexible in their behavioural routines, and continue to use dysfunctional strategies even when more efficient alternatives are available (Carroll & Rosson, 1987). To address this apparent discrepancy between rapid adaptation and rigid perseverance future studies will have to incorporate issues of learning and transfer.

The implications of this line of research are manifold:

On a theoretical and conceptual level, a strong version of the interactive perspective challenges the distinction between agent and environment, and promises to bridge the gap between cognition and action (Clark, 1997; Kirsh, 1996).

Methodologically, the dynamic interplay of factors illustrates that studies of interactive cognition have to strive for a very fine-grained resolution. To study the features of an agent, task, or task environment in isolation would fail to capture the multi-faceted nature of effects and misrepresent the complex balancing act of successful problem solving.

Finally, the study of interactive problem solving promises practical applications. Several studies have now shown that subtle changes in interactional resources can lead to substantial differences in performance. The challenge for interface design is to understand the complex structure of costs and benefits imposed by different environments, and to use this understanding to produce information displays that encourage effective interactions.

Acknowledgments

We would like to thank Suzanne Charman, Will Reader, and the anonymous reviewers for their comments and suggestions on an earlier draft. This research was supported by ESRC Research Studentship Award No. R00429934325 to HN.

References

Carroll, J.M., & Rosson, M.B. (1987). The paradox of the active user. In J.M. Carroll (Ed.), *Interfacing thought: Cognitive aspects of human-computer interaction*. Cambridge, MA: The MIT Press.

Clark, A. (1997). *Being there: Putting brain, body, and world together again*. Cambridge, MA: The MIT Press.

Gray, W.D. & Fu, W.-t. (2001). Ignoring perfect knowledge in-the-world for imperfect knowledge in-the-head: Implications of rational analysis for interface design. In *CHI Letters, 3(1)*.

Kirsh D. (1995a). Complementary Strategies: Why we use our hands when we think. In *Proceedings of the Seventeenth Annual Conference of the Cognitive Science Society*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Kirsh, D. (1995b). The Intelligent Use of Space. *Artificial Intelligence*, *73*, 31–68.

Kirsh, D. (1996). Adapting the environment instead of oneself. *Adaptive Behavior*, *4*, 415–452.

Kirsh, D., & Maglio, P. (1994). On distinguishing epistemic from pragmatic action. *Cognitive Science*, *18*, 513–549.

Larkin, J.H. & Simon, H.A. (1987). Why a diagram is (sometimes) worth ten thousand words. *Cognitive Science, 11*, 65–99.

Maglio, P.P., Matlock, T., Raphaely, D., Chernicky, B., & Kirsh D. (1999). Interactive Skill in Scrabble. In *Proceedings of the Twenty-first Annual Conference of the Cognitive Science Society* (pp. 326–330). Mahwah, NJ: Lawrence Erlbaum Associates.

Mayer, R.E. (1992). *Thinking, problem solving, cognition*. New York: W.H. Freeman.

Miller, G.A., Galanter, E., & Pribram, K. (1960). *Plans and the structure of behavior.* New York: Holt, Rinehart and Winston.

Monk, A. (1998). Cyclic interaction: a unitary approach to intention, action and the environment. *Cognition, 68*, 95–110.

Newell, A., & Simon, H.A. (1972). *Human problem solving*. Englewood Cliffs, NJ: Prentice-Hall.

O'Hara, K.P., & Payne, S.J. (1998). The effects of operator implementation cost on planfulness of problem solving and learning. *Cognitive Psychology*, *35*, 34–70.

Payne, S.J. (1991). Display-based action at the user interface. *International Journal of Man-Machine Studies, 35*, 275–289.

Siegler, R.S., & Lemaire, P. (1997). Older and younger adults' strategy choices in multiplication: Testing predictions of ASCM via the choice/no-choice method. *Journal of Experimental Psychology: General*, *126*, 71–92.

Zhang, J. (1997). The nature of external representations in problem solving. *Cognitive Science*, *21*, 179–217.

Zhang, J. & Norman, D.A. (1994). Representations in distributed cognitive tasks. *Cognitive Science*, *18*, 87–122.